Simulation of φ^4 Theory in the Strong Coupling Expansion beyond the Ising Limit

Korrigierte Version

D I P L O M A R B E I T

zur Erlangung des akademischen Grades

Dipl.-Phys.
im Fach Physik

eingereicht an der
Mathematisch-Naturwissenschaftlichen Fakultät I
Humboldt-Universität zu Berlin

von
Ingmar Vierhaus
geboren am 04.11.1983 in Kassel

Präsident der Humboldt-Universität zu Berlin:
Prof. Dr. Dr. h.c. Christoph Markschies

Dekan der Mathematisch-Naturwissenschaftlichen Fakultät I:
Prof. Dr. Andreas Herrmann

Gutachter:
1. Prof. Dr. Ulrich Wolff
2. Priv. Doz. Dr. Martin Hasenbusch

eingereicht am: 12.05.2010

Abstract

This thesis reports on the simulation of φ^4 theory using the worm algorithm, a recent simulation method which has been proven to eliminate critical slowing down in a number of statistical systems.

The untruncated strong coupling represention of the theory is derived and two variants of the worm algorithm for φ^4 theory are presented.

The algorithm is implemented in C, and we report on tests in the Ising and Gaussian limits of the theory as well as at finite coupling strengths, and compare results with a standard Metropolis simulation.

The dynamical behavior of the algorithm is examined in detail in the Gaussian case and in the interacting case at $\lambda = 0.5$ in two, three and four dimensions. We find substantial critical slowing down in the two-dimensional Gaussian case and measure critical exponents of up to 1.6. This is reduced in three dimensions where we measure critical exponents below one, and further reduced in four dimensions where critical exponents are below 0.6. We present some heuristic arguments that this is due to very long autocorrelations in the population of the link field.

In the interacting theory, we find short autocorrelations independent of the dimension of the theory. Different observables here show distinct dynamical behavior, but all measured critical exponents are below 0.55. In several cases we also find logarithmic behavior.

An approach to the estimation of the renormalized coupling using the worm algorithm is presented, but tests show that it is ill-suited for critical systems.

Zusammenfassung

Diese Arbeit beschäftigt sich mit der Simulation der φ^4-Theorie mit dem Wurm-Algorithmus, einer Simulationsmethode die sich als sehr effizient bei der Betrachtung kritischer Systeme gezeigt hat.

Die Entwicklung der Theorie in die Strong-Coupling-Expansion wird beschrieben und zwei Varianten des Wurm-Algorithmus werden vorgestellt.

Eine Implementierung des Algorithmus in C wird angefertigt und die Ergebnisse von Tests im Ising- und Gauß-Limes sowie in φ^4-Theorie mit endlicher Wechselwirkung werden präsentiert und mit einer konventionellen Metropolis-Simulation verglichen.

Anschließend wird die Dynamik des Algorithmus ausführlich im freien sowie im wechselwirkenden Fall der Theorie in zwei, drei und vier Dimensionen untersucht. In der freien zweidimensionalen Theorie weist der Algorithmus ausgeprägtes Critical Slowing Down mit kritischen Exponenten von bis zu 1.6 auf. In drei Dimensionen liegen alle kritischen Exponenten unter 1.0, und in vier Dimensionen messen wir kritische Exponenten von bis zu 0.6. Wir präsentieren eine Heuristik, die dieses Verhalten auf lange Autokorrelationen in der Besetzung des Link-Feldes zurückführt.

Unsere Messungen in der wechselwirkenden Theorie werden bei $\lambda = 0.5$ durchgeführt. Autokorrelationszeiten sind hier wesentlich kürzer als in der freien Theorie. Kritische Exponenten sind von Observable zu Observable sehr unterschiedlich, liegen aber in jedem Fall unter 0.55. In vielen Fällen wird auch ein ein logarithmischer Zusammenhang festgestellt.

Ein Methode zur Schätzung der renormierten Kopplung mit dem Wurm-Algorithmus wird vorgestellt. Tests zeigen jedoch, dass sich dieser Ansatz nicht für kritische Systeme eignet.

This thesis comes with a visualization of two runs of the Worm algorithm in the form of a flip-book on the bottom of each page. We portray simulations of φ^4 theory at $\lambda = 0.5$ on a periodic 4×4 lattice with different parameters on even and odd pages. More details will be given in Section 4.4.1.

Contents

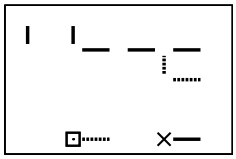
1 Introduction

In this thesis, we report on Monte Carlo (MC) simulation of φ^4 theory using a new simulation method.

MC simulations are nowadays a standard tool of elementary particle physics, as they allow for an estimation of path integrals which arise in the Euclidean formulation of field theories. To gain numerical access, a continuous theory is usually discretized on a finite space-time lattice, which provides a regularization. The continuum limit is then realized at critical points where the correlation length diverges [1]. This however poses a problem to most standard MC approaches, as they show a strong dependence of the achieved precision on the criticality of the system. Typically, the computational effort required in order to calculate a given amount of independent estimates depends on the correlation length according to a power law. This behavior is referred to as *critical slowing down* [2]. However, the dynamic behavior is a property of the algorithm and not of the simulated system.

Over the last decades there has been great effort to reformulate algorithms in order to improve their dynamic behavior. Among the successful approaches are overrelaxation [3] and multigrid [4] techniques, as well as percolation cluster methods [4, 5]. The latter have first been proposed for the Ising model, and were then generalized to $O(n)$ spin models. They introduce a second update step, which is executed not in the "natural" spin field representation of the system, but in the Fortuin Kasteleyn bond representation [6]. In systems close to criticality, this second step has the potential to effect changes on large length scales, in contrast to the standard update which consists of a change limited to a single lattice site.

In 1998, a new approach was proposed by Prokof'ev, Svistunov and Tupitsyn for quantum Monte Carlo simulations [7]. Three years later, Prokof'ev and Svistunov demonstrated its applicability to statistical systems, among them the three-dimensional Gaussian model and the two- and three-dimensional Ising models [8]. This new algorithm was given the figurative name worm algorithm. It is based on a very general idea. As a first step, the system is translated into a new representation by means of the strong coupling or hopping parameter expansion. In the new representation, the field is no longer located at lattice sites, but at links connecting each pair of neighboring sites. Also the new link field has discrete values, even if the original field was continuous. Configurations in this formulation have a graphical representation as collections of paths. Particularly, configurations containing an arbitrary number of closed paths and one open path are connected to two-point functions. The worm algorithm now allows to sample those configurations by moves that are local in the new representation. In contrast to percolation cluster methods, the original representation of the system does not play a rôle during simulation. Observables are extracted by means of new estimators from the



new representation.

In 2007, Deng and Sokal published a detailed analysis of the critical behavior of the new method in the two- and three-dimensional Ising model [9]. Their results, as well as the results published in [8, 10] show that critical slowing down in the worm algorithm is drastically reduced compared to standard algorithms. It may seem surprising that this can be achieved with an algorithm consisting exclusively of local updates, but this intuition stems from the original representation of a long-range correlated field on a lattice. The local moves in the parameter space of the strong coupling expansion do however not correspond to local moves in the spin field representation.

The idea of the worm algorithm may be generalizable to a wide range of systems, and in this respect might make good on hopes that were so far disappointed by the percolation cluster method. In this context, its applicability to systems with finite couplings is of interest.

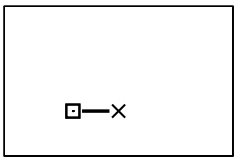
φ^4 theory is one of the simplest interacting renormalizable field theories that can be constructed. However, the Higgs sector of the electroweak theory is described by a four-component scalar field with a quartic self-interaction and interactions with other fields. Switching off the latter yields the four-component φ^4 theory [11]. Furthermore, the one-component theory already posesses a spontaneously broken symmetry, and can be used as a toy model to study this mechanism. Another application arises in the study of Bose-Einstein condensates, where the three-dimensional two-component φ^4 theory can be matched to the dilute bose gas problem [12]. It is worthwile to investigate into methods that allow for a more precise evaluation of φ^4 theory, in order to enable a better understanding of these phenomena.

In the past, φ^4 theory has also proven to be a convenient testing ground for new approaches to the numerical evaluation of field theories on the lattice. Since it is simple and well studied, it can help to gain experience with a new idea, and serve as preparation for a generalization to theories that offer a better description of nature.

Hence, the primary goal of this thesis is to provide a detailed analysis of the critical behavior of the worm algorithm in φ^4 theory. In the limit of an infinitely strong interaction, φ^4 theory passes over into the Ising model. The free theory on the other hand is identical with the Gaussian model. Since studies for the Ising limit exist [9, 13, 10], we will concentrate on the free theory and the case of finite interaction. In [9] different dynamical behavior for different observables was found and we will accordingly extend our analysis to a set of characteristic observables.

Another interesting topic is the trivialiy of φ^4 theory. A trivial theory is non-interacting after the removal of the regulator, but a regularized trivial theory can be far from free at energies below the cutoff. In fact, pure Quantum Electrodynamics and the standard $SU(2)$ Higgs model are believed to be trivial [14]. In [15] the conjecture of triviality has been used to calculate a so called triviality bound for the Higgs mass by examining the isolated scalar sector of the standard model.

φ^4 theory has been proven to be trivial in more than four dimensions [16], and there are numerous analytical and numerical results which indicate this to be the case in four dimensions as well. One contribution to the ongoing research was recently published by Wolff [10]. With analytic results found by Aizenman [17] a strong coupling estimator



for the renormalized coupling of the Ising model was formulated and implemented in the context of the worm algorithm. In standard methods, the subtraction of disconnected parts of the four-point susceptibility generally results in a loss of precision. With the new estimator, disconnected parts are subtracted analytically, resulting in achieved precisions orders of magnitude higher than in previous numerical checks on triviality.

The secondary goal of this thesis is to investigate the possiblity of the formulation of a similar estimator for the renormalized coupling in full φ^4 theory.

The structure of this thesis is the following. We will begin with the introduction of the Euclidean formulation of φ^4 theory in the second chapter. The Ising and Gaussian limits will be discussed and lattice regularization will be covered briefly.

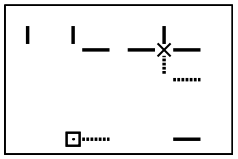
In the third chapter, we will introduce the method of Markov chain Monte Carlo simulation and outline the specifics of the involved error estimation. In this context, we will also discuss the comparison of algorithms in terms of performance.

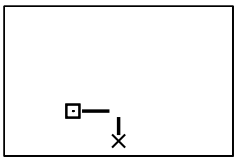
The fourth chapter contains a detailed presentation of the worm algorithm for φ^4 theory. We will derive the strong coupling expansion and describe two possible update procedures as well as the new estimators. In the fifth chapter, measurements conducted in order to test our implementation of the algorithm will be discussed.

In the sixth chapter we will analyze the critical behavior of the worm algorithm in two, three and four dimensions in the Gaussian limit as well as in the interacting theory.

In the seventh chapter, we examine the possibility to extract four-point functions from replica of two-insertion simulations.

In the last chapter, we will draw conclusions from our findings and provide an outlook.





2 φ^4 Theory on a Lattice

In this chapter, we will first point out some parallels between field theory and statistical physics. We will then derive a discretized form of the action of φ^4 theory, which allows for the exploitation of those similarities, and continue with a brief discussion of φ^4 theory on the lattice.

2.1 Path Integrals and Statistical Systems

The path integral formalism was first introduced to quantum mechanics by Feynman. It states, that for a system with Hamilton operator

$$H = \frac{p^2}{2m} + V(x), \tag{2.1}$$

transition amplitudes over a time interval $(0, T)$ can be written as [18]

$$\langle x_N | \exp(-iTH) | x_0 \rangle = \int_{x_0}^{x_N} Dx(t) \exp(iS[x(t)]). \tag{2.2}$$

The integration measure $Dx(t)$ is to be understood as the integration over all possible paths, hence the name path integral. A formal definition can be achieved by dividing the interval $(0, T)$ into a grid with points $t_i, i = 0, 1, 2, ..., N$. A path is then completely defined by locations $x(t_i)$ for all i. In the limit $N \to \infty$ the action S reads

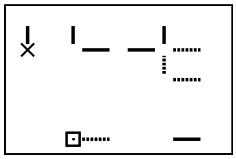
$$S = \int_0^T dt \left(\frac{m}{2} \dot{x}^2 - V(x(t)) \right). \tag{2.3}$$

This can be generalized to quantum fields with a given action S. Then, states $x(t)$ are replaced by fields $\varphi(t)$. The integration can be extended to include an arbitrary number of analogously discretized space dimensions $\varphi(t, \vec{x})$, and time and space dimensions can be written in a single vector $\varphi(t, \vec{x}) \equiv \varphi(x)$. Next, the path integral is rewritten in Euclidean time by introducing an imaginary time τ

$$t \to -i\tau. \tag{2.4}$$

After this reformulation, expectation values of observables can be written as [19]

$$\langle O \rangle = \frac{1}{Z} \int \mathcal{D}\varphi e^{-S_E[\varphi]} O[\varphi] \tag{2.5}$$



with

$$Z = \int \mathcal{D}\varphi e^{-S_E[\varphi]}. \tag{2.6}$$

The action in Minkowski space S was replaced by the Euclidean action S_E, which has to reflect the continuation of the time axis. This reformulation has the advantage of replacing the oscillating exponential in (2.2) by a dampening factor $\exp(-S_E)$. It also allows to treat time and space dimensions equivalently.
For a scalar field on a lattice with lattice sites x, the integration measure can be defined by

$$\mathcal{D}[\varphi] = \prod_x d\varphi[x] \tag{2.7}$$

and amounts to the integration over all possible field configurations.

In statistical mechanics, the expectation value of an observable O of a spin system in a heatbath with inverse temperature β is calculated by averaging over every possible configuration s weighted with the Boltzmann factor $e^{-\beta H[s]}$ [19]:

$$< \mathcal{O} >= \frac{1}{Z} \sum_{\{s\}} e^{-\beta H[s]} O[s] \tag{2.8}$$

with

$$Z = \sum_{\{s\}} e^{-\beta H[s]} \tag{2.9}$$

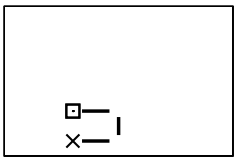
Comparing this to (2.5) we find that the Euclidean action in (2.5) can be considered the Boltzmann factor of a statistical system. Then the Integral over the continuous field values plays the rôle of the sum over spin configurations. This analog allows us to attack path integrals like (2.5) with the broad spectrum of tools developed for the analysis of statistical physics.

2.2 Discretization of φ^4 Theory

We will now proceed with the discretization of our theory. This is necessary to allow for a numerical treatment, but it also automatically provides us with a regularization scheme. The Euclidean action is defined in terms of the Euclidean Lagrangian $\mathcal{L}_E$ and reads [1]

$$S_E[\varphi] = \int d^D x \, \mathcal{L}_E = \int d^D x \left\{ \frac{1}{2}(\partial_\mu \varphi)^2 + \frac{m_0^2}{2}\varphi^2 + \frac{g_0}{4!}\varphi^4 \right\}, \tag{2.10}$$

with the scalar one component field φ, the bare mass parameter m_0, the bare strength of the self-coupling g_0 and the dimension D. We now move from continuous space to a discretized space, where the field values are located at the sites of a hypercubic lattice. The distance between neighboring sites along every axis is the lattice spacing a. This limits possible momenta to the Brillouin zone and acts as an ultraviolet regularization. The action now needs to be written in a way that retains the continuum action in



the corresponding limit ($a \to 0$). This condition is satisfied by simple next neighbor difference quotients:

$$\partial_\mu \varphi(x) \to \frac{\varphi(x + \hat{\mu}a) - \varphi(x)}{a} \tag{2.11}$$

The spacetime integral becomes a sum over lattice sites, schematically

$$\int_{-\infty}^{\infty} d^D x \to a^D \sum_x . \tag{2.12}$$

A rescaling that renders the field a dimensionless quantity

$$\varphi^2 \to \frac{a^{D-2}}{\beta} \varphi^2 \tag{2.13}$$

and a shift of the Lagrangian

$$\mathcal{L}_\mathcal{E} \to \mathcal{L}_\mathcal{E} + \lambda \tag{2.14}$$

allow us to write the action in the following convenient and completely equivalent shape:

$$S_E[\varphi] = \sum_x \left\{ \varphi(x)^2 + \lambda[\varphi(x)^2 - 1]^2 \right\} - \beta \sum_{<xy>} \varphi(x)\varphi(y) \tag{2.15}$$

$$= S_0 + S_c. \tag{2.16}$$

Here, $\sum_{<xy>}$ denotes the sum over all next-neighbor pairs on the lattice. We will use the notation S_0 for the on-site part and S_c for the coupling part later on. The relations between "old" and "new" parameters are

$$g_0 = 4! \frac{\lambda a^{D-4}}{\beta^2}, \qquad m_0^2 a^2 = (1 - 2\lambda)\frac{2}{\beta} - 2D. \tag{2.17}$$

This shape of the discretized φ^4 action is strongly reminiscent of a statistical system, with a next-neighbor coupling of strength β. In the limits of $\lambda = 0$ and $\lambda \to \infty$ it describes well known and investigated systems. This provides us with two controlled environments which we will use as sources of reference values.

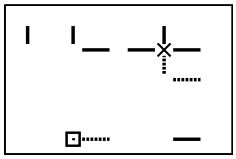
2.3 The Ising Limit $\lambda \to \infty$

To examine the action in the limit $\lambda \to \infty$, we consider the measure of the path integral (2.5) for an observable A at a single lattice site x_0 and abbreviate $\varphi(x_0)$ with φ. We define

$$f(\varphi) = \left(\varphi^2 - 1\right)^2, \qquad g(\varphi) = e^{-\varphi^2}, \qquad h(\varphi) = e^{-\varphi^2} A(\varphi), \tag{2.18}$$

and use them to write

$$\frac{\int d\varphi e^{-\varphi^2 - \lambda(\varphi^2-1)^2} A(\varphi)}{\int d\varphi e^{-\varphi^2 - \lambda(\varphi^2-1)^2}} = \frac{\int d\varphi e^{-\lambda f(\varphi)} h(\varphi)}{\int d\varphi e^{-\lambda f(\varphi)} g(\varphi)}. \tag{2.19}$$



The function f has global minima at ± 1 and a local maximum at 0. We split f, g, h into two functions, defined for positive and negative values of φ, for example

$$f(\varphi) = \begin{cases} f_-(\varphi), & \text{if } \varphi \leq 0 \\ f_+(\varphi), & \text{if } \varphi \geq 0 \end{cases}. \tag{2.20}$$

We then prepare for a saddle point expansion by substituting

$$\varphi = \frac{\eta_\pm}{\sqrt{\lambda}} \pm 1, \qquad \int_{-\infty}^{\infty} d\varphi \to \int_{-\infty}^{\sqrt{\lambda}} \frac{d\eta_-}{\sqrt{\lambda}} + \int_{-\sqrt{\lambda}}^{\infty} \frac{d\eta_+}{\sqrt{\lambda}}. \tag{2.21}$$

Expanding f around $\varphi = \pm 1$ or equivalently around $\eta_\pm = 0$ yields

$$f_\pm(\eta_\pm) = 4\frac{\eta_\pm^2}{\lambda} + O\left(\eta_\pm^3/\lambda^{3/2}\right). \tag{2.22}$$

This has the form of an expansion in $1/\sqrt{\lambda}$. We proceed similarly for g and h:

$$g_\pm = e^{-1} + O\left(\frac{\eta_\pm}{\sqrt{\lambda}}\right) \tag{2.23}$$

$$h_\pm = e^{-1}A(\pm 1) + O\left(\frac{\eta_\pm}{\sqrt{\lambda}}\right) \tag{2.24}$$

We use this to seperately evaluate numerator and denominator of (2.19) and receive

$$\int d\varphi e^{-\lambda f(\varphi)} h(\varphi) = \int_{-\infty}^{\sqrt{\lambda}} \frac{d\eta}{\sqrt{\lambda}} e^{-4\eta^2} \left(e^{-1}A(-1) + O\left(\frac{\eta}{\sqrt{\lambda}}\right)\right) \tag{2.25}$$

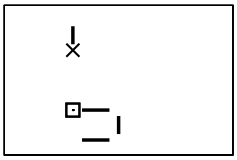
$$+ \int_{-\sqrt{\lambda}}^{\infty} \frac{d\eta}{\sqrt{\lambda}} e^{-4\eta^2} \left(e^{-1}A(+1) + O\left(\frac{\eta}{\sqrt{\lambda}}\right)\right) \tag{2.26}$$

$$\int d\varphi e^{-\lambda f(\varphi)} g(\varphi) = \int_{-\infty}^{\sqrt{\lambda}} \frac{d\eta}{\sqrt{\lambda}} e^{-4\eta^2} \left(e^{-1} + O\left(\frac{\eta}{\sqrt{\lambda}}\right)\right) \tag{2.27}$$

$$+ \int_{-\sqrt{\lambda}}^{\infty} \frac{d\eta}{\sqrt{\lambda}} e^{-4\eta^2} \left(e^{-1} + O\left(\frac{\eta}{\sqrt{\lambda}}\right)\right). \tag{2.28}$$

$$\tag{2.29}$$

The factors $\sqrt{\lambda}$ from the substitution cancel in (2.19). As $\lambda \to \infty$ the integration boundaries diverge. The exponential falls off rapidly for $\eta \to \infty$ and we can extend the integration domain in all integrals to the entire real axis. The terms of $O\left(\frac{\eta}{\sqrt{\lambda}}\right)$ vanish



in the limit for the same reason. Solving the resulting Gaussian integrals yields

$$\lim_{\lambda\to\infty} \frac{\int d\varphi e^{-\varphi^2-\lambda(\varphi^2-1)^2} A(\varphi)}{\int d\varphi e^{-\varphi^2-\lambda(\varphi^2-1)^2}} = A(1) + A(-1). \tag{2.30}$$

The measure reduces the possible field values to $\varphi(x) \in \{1, -1\}$. We call the resulting spin field $s(x)$. The constant on-site part of the action can be neglected, which reduces S_E to

$$S_E[s] = -\beta \sum_{<xy>} s(x)s(y). \tag{2.31}$$

Calculating an observable of this action by evaluating the path integral (2.5) is equivalent to calculating an observable of a statistical system with energy

$$-H = \sum_{<xy>} s(x)s(y). \tag{2.32}$$

This Hamilton function is well known from the ferromagnetic Ising model. It has been solved exactly in one and two dimensions, and for higher dimensions results from precise numerical simulations are readily available for comparison [20].

2.4 Finite Volume

In order to represent the field in computer memory, we must limit the considered space to a finite volume and choose appropriate boundary conditions. We will from here on assume periodic boundary conditions in every dimension and accordingly consider all coordinates modulo L, i. e. $x_\mu \to x_\mu(\mathrm{mod}L)$. On a one-dimensional lattice for example, this identifies the site at (L) with the site at (0). Consequently, the sites at $(L-a)$ and at (0) are next neighbors.
On a periodic lattice, there are only a finite number of possible momenta. Their components can be defined by

$$p_\mu \in \left\{0, \frac{2\pi}{L}, \frac{4\pi}{L}, ..., \frac{2\pi(L/a-1)}{L}\right\}. \tag{2.33}$$

2.5 The Gaussian Limit $\lambda = 0$

In this limit, the parameter relations (2.17) simplify to

$$g_0 = 0, \qquad m^2a^2 = \frac{2}{\beta} - 2D. \tag{2.34}$$

Since the interacting term is eliminated, we have a free scalar field on the lattice. Note that we have dropped the subscript on m_0 for this section, since in the free case, the bare mass can be considered physical. The free Euclidean propagator $G_0(x-y)$ is defined by

$$(-\partial_\mu \partial_\mu + m^2) G_0(x-y) = \delta(x-y) \tag{2.35}$$

and can be expressed as the discrete Fourier transform of the momentum-space propagator

$$G_0(z) = \frac{1}{L^D} \sum_p e^{ipz} \tilde{G}_0(p), \tag{2.36}$$

where the sum runs over all lattice momenta (2.37). Using this representation in (2.35) allows for a decomposition into Fourier modes, and yields the propagator in momentum space

$$\tilde{G}_0(p) = \frac{1}{m^2 + \hat{p}^2}, \qquad \hat{p}_\mu = \frac{2}{a} \sin(a p_\mu / 2), \tag{2.37}$$

which can then be inserted into (2.33) to give

$$G_0(z) = \frac{1}{L^D} \sum_p \frac{e^{ipz}}{m^2 + \hat{p}^2}. \tag{2.38}$$

The finite sum can easily be evaluated. Particularly, the two-point susceptibility can be calculated

$$\chi_2 \overset{\lambda=0}{=} \sum_z G_0(z) = \frac{1}{m^2}. \tag{2.39}$$

A quantity that will be convenient for testing the simulation shall be derived in the following lines, using (2.38):

$$E_I = \sum_\mu \left[L^D G_0(\mu) + \frac{2}{G_0(0)} \sum_x G_0(x) G_0(x+\mu) \right]. \tag{2.40}$$

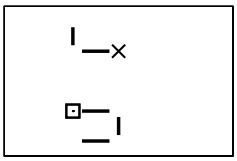
In the outer sum on the right hand side, μ runs over vectors pointing to the next lattice site in all D directions. $G_0(\mu)$ is therefore the free propagator between two neighboring sites in the μ-th direction. The left side of the equation will become meaningful later on and should at this point only be regarded as a denomination. We consider the occurring Greens functions one by one:

$$G_0(0) = \frac{1}{L^D} \sum_p \frac{1}{m^2 + \hat{p}^2} \tag{2.41}$$

$$G_0(\mu) = \frac{1}{L^D} \sum_p \frac{\cos(p_\mu)}{m^2 + \hat{p}^2} \tag{2.42}$$

and

$$\sum_x G_0(x) G_0(x+\mu) = \frac{1}{L^D} \sum_p \frac{\cos(p_\mu)}{(m^2 + \hat{p}^2)^2}. \tag{2.43}$$



Combining (2.40) with (2.41), (2.42) and (2.43) yields the final result

$$E_I = \sum_\mu \left[\sum_p \frac{\cos(p_\mu)}{m^2 + \hat{p}^2} + 2 \frac{\sum_p \frac{\cos(p_\mu)}{(m^2+\hat{p}^2)^2}}{\sum_q \frac{1}{m^2+\hat{q}^2}} \right]. \tag{2.44}$$

2.6 Renormalized Mass

The physical particle mass m is in general defined in terms of the pole of the propagator in momentum space closest to the origin. In the free theory, it is completely defined by the bare mass. With $a = 1$ the relation reads

$$m = 2 \log \left((1 + m_0^2/4)^{1/2} + \frac{m_0}{2} \right). \tag{2.45}$$

In general, it can be extracted from the decay of the partial Fourier transform C which is the Fourier transform in $D-1$ dimensions. We will call the remaining direction the time dimension. For the following lines, we denote a D-dimensional vector z also as (x,t) where x is a $D-1$-dimensional vector, and t is the time coordinate. C is defined as

$$C(t,p) \equiv a^{D-1} \sum_x e^{-ipx} G_c((x,t)) \tag{2.46}$$

with the connected two-point Greens function

$$G_c((x,t)) = G((x,t)) - \langle \varphi(x,t) \rangle \langle \varphi(0) \rangle \tag{2.47}$$

which we have defined in terms of the common two-point Greens function

$$G(z) = \langle \varphi(0) \varphi(z) \rangle . \tag{2.48}$$

The decay of the partial Fourier transform is governed by the physical mass [11]

$$C(0,t) \propto e^{-m|t|} + ... \tag{2.49}$$

where the points indicate terms that fall off faster with t. The physical mass can therefore be extracted from the asymptotic exponential decay of C.

On a finite lattice, it can be of advantage to also use another definition of the mass as in [21]. The second moment mass is defined in terms of the smallest momenta in a periodic volume:

$$\frac{m^2}{m^2 + \hat{p}_*^2} = \frac{\tilde{G}(p_*)}{\tilde{G}(0)}. \tag{2.50}$$

The smallest lattice momentum is

$$p_* = (2\pi/L, 0, 0, ..., 0), \qquad a\hat{p}_* = 2 \sin(\pi/L). \tag{2.51}$$

For $\lambda = 0$ we can write using (2.37)

$$\frac{\tilde{G}_0(p_*)}{\tilde{G}_0(0)} = \frac{m_0^2}{m_0^2 + \hat{p}_*^2} \tag{2.52}$$

and thereby identify the bare mass and the second moment mass in the free case.

In the interacting theory, we expect the Greens functions to remain finite in the continuum limit, which also holds for the second moment mass.

2.7 Renormalized Coupling

Similar to the renormalized mass, there is a counterpart to the bare coupling, the renormalized coupling. A standard definition, which is useful for simulations, is

$$g_R = -\frac{\chi_{4,c}}{\chi_2} m^D. \tag{2.53}$$

The renormalized coupling is here defined in terms of the renormalized mass m, the two-point susceptibility

$$\chi_2 = \sum_x \langle \varphi(0)\varphi(x) \rangle = \sum_x G(x) \tag{2.54}$$

and the connected four-point susceptibility

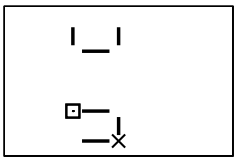
$$\chi_{4,c} = \sum_{x,y,z} \langle \varphi(0)\varphi(x)\varphi(y)\varphi(z) \rangle - 3V(\chi_2)^2. \tag{2.55}$$

The renormalized coupling is a measure for the interaction strength and vanishes in the free case.

2.8 Removing the Cutoffs

As mentioned above, the discretization of spacetime and the limitation to a finite volume introduce an infrared and an ultraviolet cutoff. As in all regularization schemes, these cutoffs must be removed to obtain physical values. Additionally, the limits must be taken in a way that keeps physics constant on the lattice.

In φ^4 theory, the two-point function falls off exponentially with the physical mass m [1]. The correlation length in physical units is given by $\xi = 1/m$. Hence, the renormalized mass can be used to fix physics on a lattice. Holding the ratio of correlation length and lattice extent constant amounts to providing $mL = \text{const}$. If this can be achieved, the limits $L \to \infty$ and $a \to 0$ can be taken. In numerical simulations, we will however always have to content ourselves with lattices of finite size and lattice spacing. Our hope is that at some point, a large and fine lattice is a good approximate. This is actually very plausible, since no laboratory is of infinite size and no experimental detector has an infinite resolution.

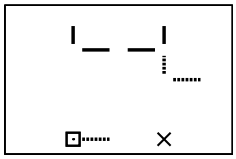


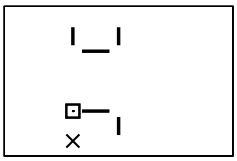
To judge the quality of the approximation, the dimensionless ratios am and Lm are meaningful. If the correlation length is small compared to the lattice size, i. e. $mL \gg 1$, and at the same time large compared to the lattice spacing, i. e. $ma \ll 1$, we expect to be close to an infinite volume continuous theory.

In the interacting case of φ^4 theory, this has to be achieved by "tuning" the parameters β and λ appropriately. In the free case, we can consider $m_0 L$ and $m_0 a$. From the parameter relations (2.34), we can conclude immediately that the continuum limit $m_0 a \to 0$ is taken by sending $\beta \to \frac{1}{D}$. This allows us to pinpoint the critical line on one end of the λ-axis:

$$\beta_{crit}(\lambda = 0) = \frac{1}{D}. \tag{2.56}$$

From the discussion above, we can derive a finite volume ultraviolet renormalization scheme which will be used in this thesis and was employed for example in [10]. We maintain a fixed ratio of lattice size and correlation length, i. e. $mL = \text{const}$, and then move towards the continuum limit by taking $L/a \to \infty$. For smaller values of mL, we expect the numerically accessible values of L/a to take us closer to the continuum limit.





3 Monte Carlo Simulation

In Chapter 2, we have outlined how path-integrals can be rewritten to make them accessible to tools of statistical physics. One of these tools is the numerical method of Monte Carlo (MC) simulation.
In statistical physics, a theory is defined by its partition function. If the sum (2.8) can be calculated, the expectation values of all observables are accessible. There are only very few cases where this has been achieved in closed form. As we have seen, in the path integrals occurring in field theory, the sum over a large but still finite number of possible configurations is replaced by a multidimensional integral over an infinite range of field values, which complicates matters further.
One possibility to make an estimation of such an integral is by means of Monte Carlo integration. The method is based on the repeated evaluation of the integrand at points in its parameter space, chosen according to a given probability distribution. As an example, we will consider the aforementioned problem

$$\bar{I} = \int \left(\prod_{i=1}^{N_x} d\varphi[x_i] \right) e^{-S_E[\varphi]} O[\varphi]. \tag{3.1}$$

Suppose $\eta_{n,x}$ are $N \times N_x$ random numbers with $n = 1, 2, .., N$ and $x = 1, 2, .., N_x$, where each set of N_x numbers is independent from the other sets and has for all n the probability distribution $p(\eta_{n,1}, \eta_{n,2}, ..., \eta_{n,N_x})$ defined on the integration domain $x \in \mathbb{R}$. Then we can write [18]

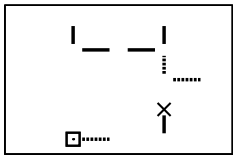
$$\lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} e^{-S_E[\varphi]} O[\varphi]\Big|_{\varphi(x_j)=\eta_{i,j}} = \int \left(\prod_{i=1}^{N_x} d\varphi[x_i] \right) p[\varphi] e^{-S_E[\varphi]} O[\varphi], \tag{3.2}$$

where the notation on the left hand side should be understood as "for every j, set $\varphi(x_j) = \eta_{i,j}$, with $j = 1, 2, ..., N_x$". The relation is true for an infinite number of evaluations of the integrand, but for a finite number of evaluations, we receive an estimate of the integral. The error of this estimate is [18]

$$\langle I \rangle = \frac{1}{N} \sum_{i=1}^{N} e^{-S_E[\varphi]} O[\varphi]\Big|_{\varphi(x_j)=\eta_{i,j}} = \bar{I} \pm \sqrt{\frac{\mathrm{var}(I)}{N}}. \tag{3.3}$$

The variance is unknown, but it can also be estimated simultaneously with the integral as follows:

$$\mathrm{var}(f) \approx \left\langle f^2 \right\rangle - \langle f \rangle^2 . \tag{3.4}$$



3.1 Importance Sampling

The distribution of our random numbers $p[\varphi]$ is so far undefined. The variance of the estimate will depend on that choice, and with it the achieved precision (3.4). We know from statistical physics that out of the large number of possible configurations of a spin system, only a relatively small fraction contributes significantly to the partition function. Also, these configurations can be located. In our example, the factor $\exp(-S_E[\varphi])$ is the equivalent to a Boltzmann factor, and the dominant part of the integrand. If we were able to sample configurations according to their probability, we could minimize the variance of the estimate. We would need random numbers distributed according to the "importance" of the configuration in the partition function. Hence the name importance sampling. To estimate the integral (3.1), we then have to change the integrand accordingly,

$$e^{-S_E[\varphi]}O[\varphi] \longrightarrow \frac{1}{p[\varphi]}e^{-S_E[\varphi]}O[\varphi]. \tag{3.5}$$

If this renders the integrand a constant for all configurations, the variance vanishes and with it the error. Unfortunately, to achieve this exactly, the exact integral would have to be known. We will therefore attempt to sample configurations according to the dominant part of the integrand.

3.2 Markov Chains

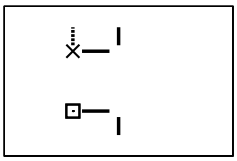
One way to create configurations distributed according to $\exp(-S_E[\varphi])$ is by use of a Markov chain. This method is conceptually different from what we have discussed so far. In Monte Carlo integration, we generate one complete configuration by drawing N_x random numbers, and then start again, discarding the previous configuration. This can be imagined as repeatedly dropping a pen over a map of phase space. In general, a better use of resources is to drop the pen once, and then begin to draw a path through phase space, which is made up of small steps. Each step then has to arise from a stochastic process, respecting the chosen probability distribution. It is possible to construct a path through phase space such that, for long paths, points in phase space are sampled with the desired Boltzmann distribution.

It is important to notice, that this comes at the cost of configurations then being correlated, influencing the expected error. This will be discussed in Section 3.4.

A Markov chain is such a path through phase space, where each configuration is derived from the previous one by means of a Monte Carlo update process [22], which is completely defined by a number of transition probabilities. We call the i-th configuration of field values in this sequence φ_i, and denote two arbitrary configurations by φ and φ'. The transition probabilities T must be positive and normalized:

$$T(\varphi_i, \varphi_{i+1}) \geq 0, \qquad \int \prod_x \mathrm{d}\varphi'(x) T(\varphi_i, \varphi') = 1. \tag{3.6}$$

Now, for the set of T to describe a valid algorithm, only two more conditions must be met



[18]: *ergodicity* and *balance*. Ergodicity is satisfied, if with a sufficient number of updates, every initial configuration can be transformed into an arbitrary new configuration. This can be written as

$$\exists n \;\; T^n(\varphi, \varphi') > 0 \;\; \forall \;\; \varphi, \varphi'. \tag{3.7}$$

The second condition, *balance*, is necessary for configurations to appear in the Markov chain with the desired probability P. Transition probabilities show balance if

$$\int \prod_x \mathrm{d}\varphi(x) T(\varphi, \varphi') P(\varphi) = P(\varphi') \tag{3.8}$$

holds. Another condition that is stronger than balance is *detailed balance*. It is satisfied if an algorithm provides

$$\frac{P(\varphi)}{P(\varphi')} = \frac{T(\varphi', \varphi)}{T(\varphi, \varphi')} \tag{3.9}$$

for all configurations φ, φ'. Stability follows directly from detailed balance:

$$\int \prod_x \mathrm{d}\varphi(x) P(\varphi) T(\varphi, \varphi') = \int \prod_x \mathrm{d}\varphi(x) P(\varphi') T(\varphi', \varphi) \tag{3.10}$$

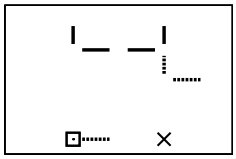
$$\overset{(3.6)}{=} P(\varphi'). \tag{3.11}$$

Detailed balance can sometimes be easier to prove than balance. Also, a combination of transitions, each satisfying detailed balance individually, again satisfies balance.

Any ergodic and balanced update procedure will, in a sufficiently long run, produce configurations φ with probability $P(\varphi)$. The first configurations will however depend on the starting configuration φ_1. The point at which this influence is negligible depends on the starting configuration itself and on the algorithm. An algorithm moving through phasespace in very small steps will remember an improbable starting configuration for a greater number of steps than a probable one from which strongly contributing configurations can be reached more quickly. In statistical physics, the latter are the equilibrium configurations and the movement from the starting configuration to "equilibrium" configurations in Monte Carlo algorithms is therefore called equilibration. Failure to equilibrate the system before using the sampled configurations for measurements will result in a bias, vanishing with the inverse chain length $1/N$ [23]. Since this bias can have a fairly large prefactor, it is dangerous to neglect it against the statistical error. In order to avoid systematic errors data should only be accumulated after the system has equilibrated.

3.3 Local Monte Carlo Algorithms

One possibility to construct update algorithms with transition probabilities as described above are local algorithms. They generate successive configurations by "local" steps. In the context of a scalar field, an example of a local step is the change of the field at a given site x_0. Transitions between configurations that differ at more than one site then



have zero transition probability. Two standard methods of constructing a local update procedure satisfying (3.7) and (3.8) will be presented in the following sections. Sites at which the update is performed can be chosen at random or in a preset order. Usually, this has no influence on performance. We will discuss changes of single field values, and therefore introduce the notation φ for an arbitrary configuration and φ'_{x_0} for a new configuration differing from φ only at x_0.

3.3.1 Heat Bath

In one heat bath step, a new field value $\varphi'(x_0)$ is assigned to the field at site x_0 with probability

$$T(\varphi, \varphi'_{x_0}) = \frac{P(\varphi'_{x_0})}{\int d\varphi'(x_0) P(\varphi'_{x_0})}. \tag{3.12}$$

This will in general only depend on a few sites, typically x_0 and its next neighbors, depending on P. Stability can be shown directly by inserting T into (3.8), and in N_x steps an arbitrary configuration can be created from any starting configuration, hence the update is ergodic. The update procedure is therefore valid.

In order to implement (3.12), it is necessary to generate random numbers with the one-dimensional distribution P.

In a system with a discrete phase space, like the Ising model, the integrals become sums. Transition probabilities for a single spin flip then take the form

$$T(\varphi, \varphi'_{x_0,\pm}) = \frac{P(\varphi'_{x_0,\pm})}{P(\varphi'_{x_0,+}) + P(\varphi'_{x_0,-})}. \tag{3.13}$$

3.3.2 Metropolis

In a Metropolis update step, a local change is proposed and accepted with a certain probability [24]. In our example, these acceptance probabilities are defined as:

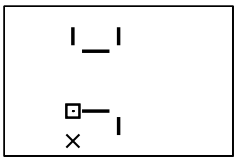
$$t(\varphi, \varphi'_{x_0}) = \min\left(1, P(\varphi'_{x_0})/P(\varphi)\right) \tag{3.14}$$

They differ from the transition probabilities T by the proposition density, i. e. the probability to propose a certain new configuration. Given a constant proposition density $1/\delta$, the transition probabilities satisfy detailed balance:

$$\frac{T(\varphi'_{x_0}, \varphi)}{T(\varphi, \varphi'_{x_0})} = \frac{\min\left(1, P(\varphi'_{x_0})/P(\varphi)\right)}{\min\left(1, P(\varphi)/P(\varphi'_{x_0})\right)} = \frac{P(\varphi)}{P(\varphi'_{x_0})} \tag{3.15}$$

The right hand side can be verified by considering the cases $P(\varphi) > P(\varphi'_{x_0})$ as well as $P(\varphi) < P(\varphi'_{x_0})$ independently. With the results from the last subsection, we conclude that a balanced update can be constructed from a sequence of metropolis-type steps.

In order to prove ergodicity of a single Metropolis step, the procedure generating the proposal for the new configuration must be specified. In the next section, we will discuss one possible proposition procedure and show its ergodicity.



3.3.3 Standard Metropolis Simulation of φ^4 Theory

As an example of a Metropolis simulation, we will here outline a standard Metropolis algorithm for φ^4 theory, as described in [25]. A version of the simulation was implemented in C in the course of the thesis and used as a source of reference values. It samples configurations with probabilities according to the reparametrized action (2.15).

A configuration consists of the set of scalar values $\{\varphi(x)\}$ at all sites x. The update proposes to change $\varphi(x_0)$ by a randomly chosen value,

$$\varphi(x_0) \to \varphi(x_0) + \Delta\varphi, \qquad \Delta\varphi \in \{-\delta, \delta\} \tag{3.16}$$

where δ should be chosen to avoid too slow movement through phase space (δ too small) as well as too small acceptance ratios (δ too high). This can be ensured by monitoring the acceptance probability. Values between 0.3 and 0.7 are usually close to optimum [25]. The change of the field implies a change of the action,

$$\Delta S(\varphi, \Delta\varphi) = S(\varphi + \Delta\varphi) - S(\varphi) \tag{3.17}$$

$$= \Delta\varphi \left((2\varphi + \Delta\varphi) \left(1 - 2\lambda + \lambda \left((\Delta\varphi)^2 + 2\varphi\Delta\varphi \right) \right) - \beta \sum_{<x_0 y>} \varphi(y) \right), \tag{3.18}$$

where $\varphi(x_0)$ was substituted with φ for brevity and the sum collects the values of all fields at sites neighboring x_0. The Metropolis acceptance probability of this update is simply

$$p_{\varphi \to \varphi + \Delta\varphi} = \min(1, \exp(-\Delta S)). \tag{3.19}$$

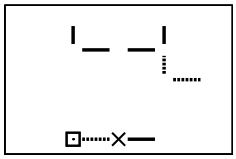
As shown in the last section, Metropolis type updates satisfy balance. The range of field values reachable in one step depends on the value of δ. The transformation of a given starting configuration $\varphi(x)$ into a given new configuration $\varphi'(x)$ in

$$n_{steps} = \sum_i \left\lceil \frac{|\varphi(x_i) - \varphi'(x_i)|}{\delta} \right\rceil \tag{3.20}$$

steps has nonzero probability if the sites at which to perform the update are selected at random. The algorithm is stable and ergodic, and hence it samples configurations weighted with their corresponding Boltzmann factor. The estimation of the integral (3.1) is thereby reduced to the "measurement" of $O[\varphi]$ for configurations φ created from the update process. Between measurements, one update is performed at each site. This sequence of N_x updates is called one *sweep*.

From the sampled configurations, estimates of observables are extracted in the natural way. Two-point functions are averaged over the volume,

$$G(z) = \frac{1}{V} \left\langle \sum_x \varphi(x)\varphi(x+z) \right\rangle. \tag{3.21}$$



Susceptibilities can be estimated from powers of the magnetization

$$M = \left\langle \sum_x \varphi(x) \right\rangle \tag{3.22}$$

$$M^2 = \left\langle \left(\sum_x \varphi(x) \right) \left(\sum_y \varphi(y) \right) \right\rangle \tag{3.23}$$

$$= V \left\langle \sum_x \varphi(0)\varphi(x) \right\rangle = V\chi_2. \tag{3.24}$$

In particular with

$$\left\langle M^4 \right\rangle = \left\langle \left(\sum_w \varphi(w) \right) \left(\sum_x \varphi(x) \right) \left(\sum_y \varphi(y) \right) \left(\sum_z \varphi(z) \right) \right\rangle \tag{3.25}$$

$$= V \left\langle \sum_{x,y,z} \varphi(x)\varphi(y)\varphi(z)\varphi(0) \right\rangle, \tag{3.26}$$

the connected four-point susceptibility (2.55) can be written as

$$\chi_4^{\text{conn}} = \frac{1}{V} \langle M \rangle^4 - \frac{3}{V} \left\langle M^2 \right\rangle^2. \tag{3.27}$$

In order to calculate the second moment mass, Greens functions in momentum space are measured according to (2.36),

$$\tilde{G}(p) = \langle \varphi(p)\varphi(-p) \rangle = \left\langle |\varphi(p)|^2 \right\rangle = a^{2D} \left\langle |\sum_x e^{-ipx}\varphi(x)|^2 \right\rangle \tag{3.28}$$

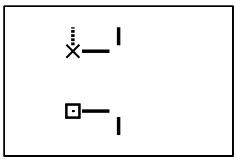
$$= a^{2D} \left\langle \left(\sum_x \cos(px)\varphi(x) \right)^2 \right\rangle + a^{2D} \left\langle \left(\sum_x \sin(px)\varphi(x) \right)^2 \right\rangle, \tag{3.29}$$

with the special case

$$\tilde{G}(0) = a^{2D} \left\langle \left(\sum_x \varphi(x) \right)^2 \right\rangle \overset{(3.24)}{=} a^{2D} V \chi_2. \tag{3.30}$$

3.4 Autocorrelation and Consequences for Error Estimation

So far, we are able to estimate observables of interest and are certain to find the exact answer for an infinite number of sampled configurations. To make estimates from a finite sampling process meaningful, an error analysis incorporating the autocorrelation of estimates is essential. We will here outline the Γ-Method of error estimation as presented in [26], which will be used to estimate errors throughout this thesis.



We consider a simulation generating N configurations after equilibration according to a generic Monte Carlo process with transition probabilities T. We write the i-th estimate for the primary observable α as a_α^i. The mean of observable α is defined as

$$\bar{a}_\alpha = \frac{1}{N}\sum_{i=1}^{N} a_\alpha^i. \tag{3.31}$$

For the difference between estimated means $\bar{a}_\alpha$ and the exact means A_α, we define

$$\bar{\delta}_\alpha = \bar{a}_\alpha - A_\alpha. \tag{3.32}$$

For large enough N_r, δ_α^r is unbiased and normally distributed, independently of the distribution of the individual estimates a_α^i, which we assume to be the case. The expected deviation of the primary observable α is given in terms of its *covariance*:

$$\left\langle \left(\bar{\delta}_\alpha\right)^2 \right\rangle = \frac{1}{N^2}\sum_{i,j=1}^{N} \left\langle (a_\alpha^i - A_\alpha)(a_\alpha^j - A_\alpha) \right\rangle = \frac{1}{N^2}\sum_{i,j=1}^{N} \Gamma_{\alpha\alpha}(j-i). \tag{3.33}$$

The autocorrelation or Γ-function of observable α defined in the last equation is the key quantity to the estimation of errors of Markov chain MC results. It relates deviations of estimates at different times of the sampling process. $\Gamma_{\alpha\alpha}(0)$ is the variance of observable α, which is a static quantity and depends only on the simulated system. For $j \neq i$, the autocorrelation is a dynamic quantity depending on the transition probabilities T, which are a property of the algorithm defining the update procedure. The autocorrelation function usually decays exponentially, and we call the timescale of the decay τ. It is useful to define the exponential autocorrelation time of observable α as in [23] by

$$\tau_{exp,\alpha} = \lim_{t\to\infty} \sup \frac{t}{-\log|\Gamma_{\alpha\alpha}(t)|}, \tag{3.34}$$

as well as the relaxation time of the "slowest mode"

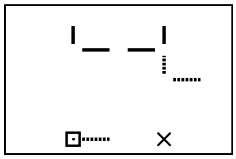
$$\tau_{exp} = \sup_\alpha \tau_{exp,\alpha}. \tag{3.35}$$

Assuming the exponential decay of $\Gamma_{\alpha\alpha}$ and a sufficiently long run $N \gg \tau$, we can eliminate the dependence of the error on the run length N by writing

$$\left\langle \left(\bar{\delta}_\alpha\right)^2 \right\rangle = \frac{1}{N^2}\sum_{t=-(N-1)}^{N-1} (N-|t|)\,\Gamma_{\alpha\alpha}(t) \tag{3.36}$$

$$= \frac{1}{N}\sum_{t=-\infty}^{\infty} \Gamma_{\alpha\alpha}(t)(1+O(\tau/N)) \tag{3.37}$$

$$\approx \frac{2\tau_{int,\alpha}}{N}\Gamma_{\alpha\alpha}(0) = \sigma_\alpha^2. \tag{3.38}$$



The error of observable α is now defined in terms of its static variance $\Gamma_{\alpha\alpha}(0)$ and its integrated autocorrelation time $\tau_{int,\alpha}$. The result has an illustrative interpretation: The integrated autocorrelation time is half the time needed to produce two independent configurations. The factor $1/2$ in the definition of $\tau_{int,\alpha}$ is convention. For $t \gg \tau$, it provides $\tau_{int,\alpha} = \tau_{exp,\alpha}$, given $\Gamma_{\alpha\alpha}$ behaves like $\exp(-t/\tau)$. We recover the error of a Monte Carlo integration (3.3) in the case of completely uncorrelated configurations $\tau_{int,\alpha} = 1/2$.

The influence of the used update procedure is now encapsulated in the integrated autocorrelation time which can be considered a measure for the performance of an algorithm. Different observables can have very different autocorrelation times, although in general $\tau_{exp} = \tau_{exp,\alpha}$ holds for all observables [23]. One can think of the observables having different amplitudes of "overlap" with the system's slowest mode of relaxation τ_{exp}.

Before we proceed, we will slightly generalize our setup to include R replica of the same simulation, each with independent random numbers and initial configuration, and encompassing N_r measurements. The respective estimates are equipped with an additional index, so that we write the i-th estimate of observable α in replicum r as $a_\alpha^{i,r}$. Equation (3.31) becomes the per replicum mean

$$\bar{a}_\alpha^r = \frac{1}{N_r} \sum_{i=1}^{N_r} a_\alpha^{i,r}, \tag{3.39}$$

and the mean of all measurements reads

$$\bar{\bar{a}}_\alpha^r = \frac{1}{N} \sum_{r=1}^{R} N_r \bar{a}_\alpha^r, \qquad N = \sum_{r=1}^{R} N_r, \tag{3.40}$$

with the total number of estimates N. The difference between overall mean and exact mean is written analogously to (3.32) as

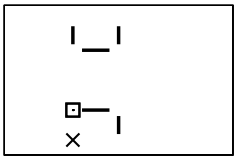
$$\bar{\bar{\delta}}_\alpha^r = \bar{\bar{a}}_\alpha^r - A_\alpha. \tag{3.41}$$

Incorporating multiple replica into the error analysis has a number of advantages. Primarily, it allows for consistency checks of the simulated data. It can also be used to improve the precision of error estimation. In the course of this theses, we have generally simulated several replica, and made use of the possible improvements in our data analysis.

3.4.1 Derived Quantities

In the course of the thesis we will frequently need to estimate errors of derived quantities F, which are a function of the primary observables

$$F \equiv f(A_1, A_2, ...) \equiv f(A_\alpha). \tag{3.42}$$



One example of such a function is already given by (3.27), where the connected four-point susceptibility is calculated from powers of the magnetization. The respective estimators for F read

$$\bar{\bar{F}} = f(\bar{\bar{a}}_\alpha), \qquad \bar{F} = \frac{1}{N}\sum_{r=1}^{N_r} f(\bar{a}_\alpha^r). \tag{3.43}$$

We generalize our notation to include off-diagonal elements of the covariance matrix

$$\left\langle \bar{\delta}_\alpha^r \bar{\delta}_\beta^s \right\rangle = \frac{1}{N_r} \underbrace{\sum_{t=-\infty}^{\infty} \Gamma_{\alpha\beta}(t)}_{C_{\alpha\beta}} \delta_{rs}\,(1 + O(\tau/N_\tau)). \tag{3.44}$$

We have made the same assumptions regarding $\Gamma_{\alpha\beta}$ as before for $\Gamma_{\alpha\alpha}$. For sufficiently accurate deviations of the primary estimators, a Taylor expansion is justified

$$\bar{\bar{F}} = F + \sum_\alpha f_\alpha \bar{\bar{\delta}}_\alpha + O\left((\max_\alpha \delta_\alpha)^2\right), \qquad f_\alpha = \frac{\partial f}{\partial A_\alpha}. \tag{3.45}$$

Writing out the next order shows a bias of the estimators $\bar{\bar{F}}$ of order $C_{\alpha\beta}/N$, which will be much smaller than the statistical error for a large enough number of independent estimates. The bias can be canceled if more than one replicum is available. This also provides a consistency check, since a large bias suggests an insufficient runlength [26]. Inserting (3.44) into (3.45), we read off the error of $\bar{\bar{F}}$ in leading order

$$\sigma_F^2 = \frac{C_F}{N} + O\left(N^{-2}\right), \qquad C_F = \sum_{\alpha\beta} f_\alpha f_\beta C_{\alpha\beta}. \tag{3.46}$$

We introduce the static variance and integrated autocorrelation time of the derived quantity F

$$v_F = \sum_{\alpha\beta} f_\alpha f_\beta \Gamma_{\alpha\beta}(0), \tag{3.47}$$

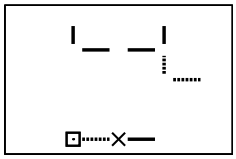
$$t_{int,F} = \frac{1}{2v_F} \sum_{t=-\infty}^{\infty} \sum_{\alpha\beta} f_\alpha f_\beta \Gamma_{\alpha\beta}(t), \tag{3.48}$$

and write the error analogously to (3.38)

$$\sigma_F^2 = \frac{2\tau_{int,F}}{N} v_F. \tag{3.49}$$

3.4.2 Γ-Method of Error Estimation

The formulas derived so far depend on integrated autocorrelation times, which are unknown. With the Γ-Method, the integrated autocorrelation time is estimated, allowing for an estimation of the error. The estimator of $\Gamma_{\alpha\beta}$ is rewritten in terms of the estimated



means

$$\bar{\bar{\Gamma}}_{\alpha\beta}(t) = \frac{1}{N - Rt} \sum_{r=1}^{R} \sum_{i=1}^{N_r - t} \left(a_\alpha^{i,r} - \bar{\bar{a}}_\alpha \right) \left(a_\beta^{i+t,r} - \bar{\bar{a}}_\beta \right). \tag{3.50}$$

For derived quantities, a projected autocorrelation can be estimated, with derivatives evaluated at the estimated means $\bar{\bar{a}}_1, \bar{\bar{a}}_2$, in the following way:

$$\bar{\bar{\Gamma}}_F(t) = \sum_{\alpha\beta} \bar{\bar{f}}_\alpha \bar{\bar{f}}_\beta \bar{\bar{\Gamma}}_{\alpha\beta}(t). \tag{3.51}$$

All quantities necessary to estimate an error are defined in terms of $\Gamma_F(t)$. We use

$$\bar{\bar{v}}_F = \bar{\bar{\Gamma}}_F(0) \tag{3.52}$$

and

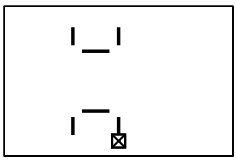
$$\bar{\bar{C}}_F(W) = \left[\bar{\bar{v}}_F + 2 \sum_{t=1}^{W} \bar{\bar{\Gamma}}_F(t) \right]. \tag{3.53}$$

The parameter W defines a summation window used to truncate the autocorrelation sum, which should be chosen carefully. A summation over the whole measured timescale is not practicable, since noise in Γ usually becomes excessive for large t. Too small summation windows on the other hand will truncate a non-negligible contribution to τ_{int}.

In the present thesis, error analyses were conducted with Matlab (2007a, The MathWorks), using the Script *UWerr.m* provided with [26]. It implements the Γ-Method of error estimation as described here, but with a number of improvements and consistency checks that were not mentioned in this brief presentation, as well as an estimation of the error of the error. Another feature is an automatic windowing procedure, and the possibility to visually check the summation window W, which was done for all datasets presented in this thesis.

3.5 Critical Slowing Down and Ways Out

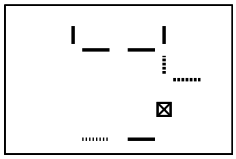
In Section 2.8 we discussed how taking the continuum limit amounts to approaching a critical point of the theory, where the correlation length ξ diverges. For a finite lattice close enough to criticality, the lattice size limits the correlation length and we can assume $\xi \sim L$. A heuristic idea of the local Monte Carlo update as a random walk through phase space would suggest, that the information given by an update step has to travel a distance of order ξ for the configuration to become decorrelated from the previous one, and one would expect $\tau \sim L^2$ [23]. This is a surprisingly good guess. Typical local algorithms like the Metropolis algorithm for φ^4 theory described in Section 3.3.3 indeed have critical exponents of $z \approx 2$ near the critical point [2]. One sweep is associated with a computational effort of order L^D. It follows that the computational effort for a given number of independent configurations grows with the correlation length as ξ^{D+z}.
This growth of the autocorrelation time with the correlation length is called *critical*

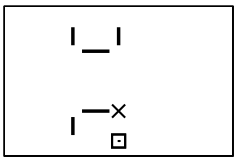


slowing down. As we have seen, the dynamical exponent z is a property of the algorithm, not of the physical system it simulates, so a better algorithm could reduce or eliminate the problem. For the Ising model, the group of cluster algorithms [27] were successful in reducing critical slowing down. The first cluster algorithm presented was the Swendsen-Wang algorithm [5], for which critical exponents around 0.5 were measured. These algorithms are however not based on local steps, but on steps encompassing a collective of sites, the clusters.

A different approach to eliminating critical slowing down, are Multigrid Monte Carlo methods. They attempt to counteract long autocorrelations by interspersing local updates on a "fine grid" with updates on a "coarse grid", which approximates the systems behavior at longer lengths scales [23]. A Multigrid Monte Carlo method for φ^4 theory was developed by Goodman and Sokal [4]. For the Gaussian model, they were able to prove that the autocorrelation is bounded as the system approaches criticality, which is equivalent to the elimination of critical slowing down. For the interacting theory, they were able to improve performance by a factor of roughly ten compared to standard methods. The critical exponent of their algorithm however is equal to that of a standard local heat bath. To our knowledge, Multigrid Monte Carlo is at present among the most efficient algorithms for φ^4 theory at finite interaction strengths.

As mentioned in the introduction, since about a decade a new approach has emerged [8], and is the subject of this thesis. In this approach, the phase space of the system is reparametrized by means of the strong coupling expansion. The algorithm than transforms graphs into each other by means of local steps in the new configuration space. The method is relatively young, and there are a number of open questions, regarding its generalizability and applicability to several theories. In this thesis, we hope to answer some of them by applying the method to φ^4 theory.





4 Worm Algorithm for φ^4 Theory

We will now describe how to translate the scalar field defined by the discretized action (2.15) into an equivalent system of links by expanding the path integral into a sum of strong coupling graphs. A Monte Carlo algorithm that samples configurations in this representation, the worm or PS algorithm, was introduced 2001 by Prokof'ev and Svistunov [8]. Along with the new representation come new estimators for physical observables, which will be discussed in Chapter 4.6. We assume from now on a periodic, hypercubic and D-dimensional lattice of extent L with unit lattice spacing. Such a lattice consists of $N_x = L^D = V$ sites, and there are $N_l = N_x D$ pairs of nearest neighbors.

4.1 The Strong Coupling Expansion

As a starting point, we consider the path integral representation of the n-point function as in (2.5)

$$\langle\varphi(x_1)\varphi(x_2)...\varphi(x_n)\rangle = \frac{\prod_x \int d\varphi(x) e^{-S[\varphi]}\varphi(x_1)\varphi(x_2)...\varphi(x_n)}{\prod_x \int d\varphi(x) e^{-S[\varphi]}} = \frac{Z(x_1, x_2, ..., x_n)}{Z(\emptyset)}. \quad (4.1)$$

From here on, we consider $Z(x_1, x_2, ..., x_n)$ of which $Z(\emptyset)$ is a special case. We plug in the action (2.15) and write the sum over pairs of neighbors as a product of exponentials and then write each exponential as a power series

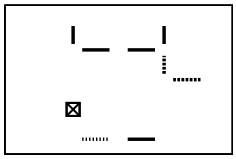
$$Z(x_1, ..., x_n) = \prod_x \int d\varphi(x) e^{-S_0[\varphi]} \left(\prod_{l=<y_1 y_2>} e^{\beta\varphi(y_1)\varphi(y_2)} \right) \varphi(x_1)...\varphi(x_n) \quad (4.2)$$

$$= \prod_x \int d\varphi(x) e^{-S_0[\varphi]} \left(\prod_{l=<y_1 y_2>} \sum_{k(l)=0}^{\infty} \frac{[\beta\varphi(y_1)\varphi(y_2)]^{k(l)}}{k(l)!} \right) \varphi(x_1)...\varphi(x_n). \quad (4.3)$$

The summation variable for each pair of neighbors l is $k(l)$. If we expand the product of sums $\prod_l \sum_k$, we obtain an infinite sum of products with l factors. Every term represents one distinct possibility to choose a value $k(l)$ for each l. We can therefore reorder sum and product

$$\prod_{l=<y_1 y_2>} \sum_{k=0}^{\infty} \longrightarrow \sum_{\{k\}} \prod_{l=<y_1 y_2>} \quad (4.4)$$

where we have denoted the summation over every possible configuration of k by $\sum_{\{k\}}$. The product over fields at different sites can now be reordered and the field insertions



can be incorporated. This yields

$$\left(\prod_{l=<y_1y_2>}[\varphi(y_1)\varphi(y_2)]^{k(l)}\right)\varphi(x_1)...\varphi(x_n) = \left(\prod_x \varphi(x)^{\sum_{l,\partial l\ni x}k(l)}\right)\varphi(x_1)...\varphi(x_n) \quad (4.5)$$

$$= \prod_x \varphi(x)^{d(x)}. \quad (4.6)$$

We have used the abbreviation

$$d(x) = \underbrace{\sum_{l,\partial l\ni x} k(l)}_{\equiv d_0(x)} + \sum_{m=1}^{n}\delta_{x_m,x}. \quad (4.7)$$

Here, $\sum_{l,\partial l\ni x}$ sums over the neighbors of site x, so that $d(x)$ is the sum of all link values $k(l)$ connecting to site x plus the number of times this site appears in $x_1, ..., x_n$.
After reinserting (4.6) into (4.3), merging the products of sites from S_0 and S_C and writing out the action, we arrive at

$$Z(x_1, x_2, ..., x_n) = \sum_{\{k\}} w(k)\prod_x c(d(x)), \quad (4.8)$$

with the definitions

$$c(\lambda, d(x)) = \int_{-\infty}^{\infty} d\varphi(x) e^{-\varphi(x)^2-\lambda[\varphi(x)^2-1]^2}\varphi(x)^{d(x)} \quad (4.9)$$

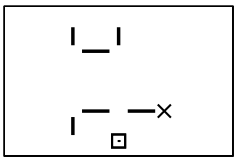
$$w(k) = \prod_l \frac{\beta^{k(l)}}{k(l)!}. \quad (4.10)$$

The integral over the scalar field can now be computed. In Section 4.5 an efficient way of obtaining precise values for the ratios of c that are sufficient for simulations is proposed. By carrying out the integral, we have effectively translated the scalar field $\varphi(x)$ located on lattice sites x into an integer link field $k(l)$ located on the links between the sites l. The complete sum over configurations of the link field is maintained, hence this is an exact reformulation.

The dependence of c on λ will be left implicit from here on.

4.2 Constraints to the Link Configuration

Taking a closer look at (4.9), we notice the symmetry of the exponential. If $d(x)$ is odd, then $\varphi(x)^{d(x)}$ is antisymmetric, and $c(d(x))$ equals zero. Field configurations that have non-zero probability must supply an even value for $d(x)$ at every site x.



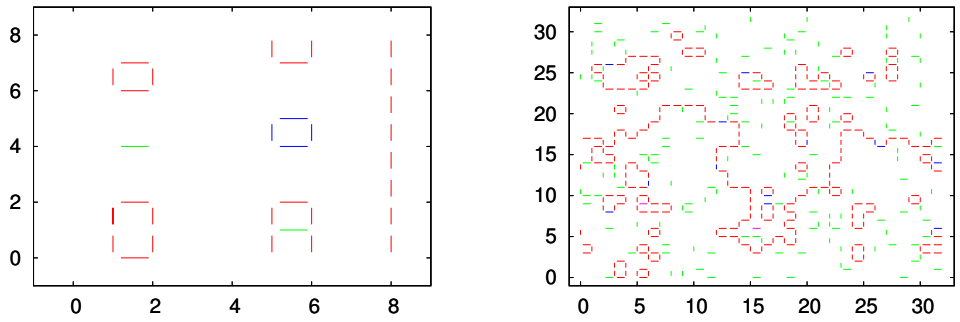


Figure 4.1: Left: Collection of simple closed paths on a periodic 8x8 lattice. Right: CP-configuration taken from a simulation of the Ising Model on a periodic 32×32 lattice. Colors correspond to link values $k(l)$: (red,green,blue) $\widehat{=}$ $(1,2,3)$.

For the proper partition function $Z(\emptyset)$, we have

$$d(x) = d_0(x) = \sum_{l,\partial l \ni x} k(l). \tag{4.11}$$

Consequently, in a valid configuration, the sum of linkvalues incident at each site must be even. In [8], this is called a closed path (CP) configuration, since there are no "loose ends". The configuration can be considered a collection of closed paths allowing for intersections. We will also refer to these configurations as vacuum graphs, due to their connection to the "vacuum" partition function. Examples of such configurations are represented in Figure 4.1.

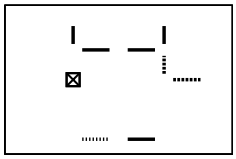
If an n-point function $Z(x_1,...,x_n)/Z(\emptyset)$ is considered, the field insertions at sites $x_1,...,x_n$ have to be taken into account. The constraint can then be expressed as the coincidence of the following two sets [17]. The source set Q containing all sites surrounded by an odd number of links

$$Q[k] = \left\{ x | \sum_{l,\partial l \ni x} k(l) = 1(\mathrm{mod}2) \right\}, \tag{4.12}$$

and the insertion set X, containing all sites that are inserted an odd number of times

$$X = \left\{ x | \sum_{i=1}^{n} \delta_{x,x_i} = 1(\mathrm{mod}2) \right\}. \tag{4.13}$$

The constraint forces these two sets to coincide. An odd number of insertions at a given site forces the sum of all values of links connecting to that site to be odd.



4.3 Enlarging the Link Ensemble

We will now consider the calculation of a two-point function $\langle\varphi(x_1)\varphi(x_2)\rangle$ from the translated partition functions. According to (4.1), we need to calculate the quotient of $Z(x_1, x_2)$ and $Z(\emptyset)$.

If $x_1 \neq x_2$, then $Z(x_1, x_2)$ contains only configurations that have one path ending at each of these sites and closed paths at all other sites. Examples of such configurations are shown in plots (3) and (4) of Figure 4.2. In the case $x_1 = x_2$ only closed path configurations are allowed in $Z(x_1, x_2)$, because then $X = \emptyset$ and the constraint requires $K = \emptyset$. These are also the configurations contained in $Z(\emptyset)$. Using (4.8) and (4.7), we can relate partition functions by writing:

$$Z(x_1, x_1) = \sum_{\{k\}} w(k) \left(\prod_{x \neq x_1} c(d_0(x)) \right) c(d_0(x_1) + 2) \tag{4.14}$$

$$Z(\emptyset) = \sum_{\{k\}} w(k) \left(\prod_{x \neq x_1} c(d_0(x)) \right) c(d_0(x_1) + 2) \frac{c(d_0(x_1))}{c(d_0(x_1) + 2)}. \tag{4.15}$$

For convenience we will from now on use the definition

$$\tilde{c}(d(x)) = \frac{c(d(x) - 2)}{c(d(x))}. \tag{4.16}$$

$Z(x_1, x_2)$ and $Z(\emptyset)$ can be calculated in one simulation if we enlarge the ensemble (4.8) by defining a new partition function

$$\mathcal{Z} = \sum_{u,v} Z(u, v). \tag{4.17}$$

In the enlarged ensemble, observables are then averaged as follows:

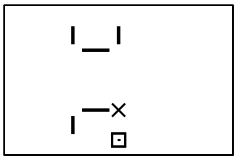
$$\langle\langle A(k, u, v)\rangle\rangle = \frac{1}{\mathcal{Z}} \sum_{u,v,\{k\}} w(k) \prod_x c(d(x)) A(k, u, v). \tag{4.18}$$

We recover $Z(x_1, x_2)$ by calculating expectation values of delta functions:

$$Z(x_1, x_2) = \mathcal{Z} \left\langle\langle \delta_{x_1,u} \delta_{x_2,v} \rangle\right\rangle. \tag{4.19}$$

Whenever the field insertions coincide, we can extract a contribution to $Z(\emptyset)$. The contribution of the insertions can be eliminated by including $\tilde{c}$ in our estimator as shown in (4.15)

$$Z(\emptyset) = \frac{\mathcal{Z}}{V} \left\langle\langle \tilde{c}(d(u)) \delta_{u,v} \rangle\right\rangle. \tag{4.20}$$



Combining (4.19) and (4.20) yields the two point function in the new ensemble

$$\langle\varphi(x_1)\varphi(x_2)\rangle = V\frac{\langle\langle\delta_{x_1,u}\delta_{x_2,v}\rangle\rangle}{\langle\langle\tilde{c}(d(u))\delta_{u,v}\rangle\rangle}. \tag{4.21}$$

4.4 Sampling the Enlarged Ensemble

Now all we need in order to calculate two-point functions is an algorithm that allows us to sample configurations according to (4.17). The PS algorithm performs this task very efficiently and in a very simple way. Configurations are updated by moving either u or v to one of its neighboring sites and changing the value of the used link in the process, thereby changing the link configuration $\{k\}$. One can imagine u as the *head* and v as the *tail* of a worm consisting of active links, i. e. links with non-zero value. As the configuration is updated, the worm moves over the lattice. This image motivates the name worm algorithm. The algorithm was originally presented in [8]. In [10] and [13], variants of that algorithm were proposed for the Ising case, one using a Metropolis and one a heat bath update. We will describe here the generalization of these algorithms to the full range of φ^4 theory.
Starting point is always an arbitrary configuration $(u, v, \{k\})$ satisfying the constraint. A convenient choice is the CP-configuration $k(l) = 0$ and random $u = v$. Before measurements can be accumulated, an appropriate thermalisation process has to take place.

4.4.1 Metropolis Update

The update consists of two elementary steps:

I We choose with equal probability one of the $2D$ neighbor sites of u and call it u'. The link connecting the two sites will be called $\hat{l}$. We propose to change the configuration as follows,

$$u \to u' \tag{4.22}$$

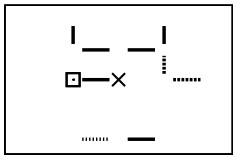
$$k(\hat{l}) \to k(\hat{l}) \pm 1, \tag{4.23}$$

choosing incrementation and decrementation of $k(\hat{l})$ with equal probability. The Metropolis acceptance probabilities can be read off from (4.8). We use the definition of $\tilde{c}$ in (4.15) for the occurring ratios of weights c to write them as

$$p_{I,+} = \min\left(1, \frac{\beta}{k(l)+1}\frac{1}{\tilde{c}(d(u')+2)}\right) \tag{4.24}$$

$$p_{I,-} = \min\left(1, \frac{k(l)}{\beta}\tilde{c}(d(u))\right). \tag{4.25}$$

This step is the movement of the worm on the lattice and will be the only step executed in the majority of cases. The update ensures positive or zero



values $k(l)$ since the probability to decrease a bond with $k(l) = 0$ is zero.

II The second step is only executed if in the current configuration u and v are on the same lattice site. From the V sites, we randomly choose one and call it u'. We propose to change the configuration by setting $u \to u', v \to u'$ and accept with probability

$$p_{II} = \min\left(1, \frac{\tilde{c}(d(u))}{\tilde{c}(d(u')+2)}\right). \tag{4.26}$$

One can imagine this step as a jump of head and tail of the worm to a new site. The values of the links $k(l)$ are not changed in this step.

One update consists of the successive execution of these two steps. An example of three successive updates is shown in Figure 4.2.
The steps satisfy detailed balance independently, so that one update satisfies balance. An algorithm consisting of these updates also shows ergodicity, since every configuration can be disassembled into the trivial configuration, and from there, every possible configuration can be constructed.
We call a group of N_x updates one iteration. The cost of one iteration is proportional to the volume, since the cost of one update is of $O(1)$. One iteration can be compared to one sweep of standard algorithms.

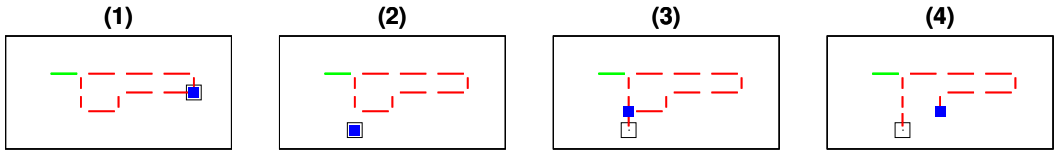
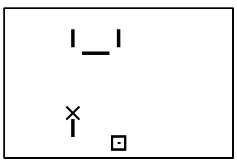


Figure 4.2: Example of configurations, that can be generated successively with one Metropolis type update in between. Squares show the locations of u and v, colors represent link values: red $\hat{=}$ 1, green $\hat{=}$ 2. In the update (1) $\to$ (2) only the II-step was accepted. In updates (2) $\to$ (3) and (3) $\to$ (4) the I-step was accepted, and the II-step was not proposed, since the configurations did not provide $u = v$. The update (2) $\to$ (3) increments and the update (3) $\to$ (4) decrements the used link. (1) and (2) are CP-configurations.

To further illustrate the movement through the new configuration space we show a succession of configurations on a periodic 4×4 lattice generated with the Metropolis update as a flip book on the bottom of each page. Both simulations were conducted in the interacting theory at $\lambda = 0.5$. On the odd numbered pages we show a run with $\beta = 0.7$ for which we have measured separately $mL = 1.0(1)$. On the even numbered pages, a simulation with $\beta = 0.5$ ($mL = 3.0(1)$) is visualized. Both systems were equilibrated beforehand. Since this thesis only has enough pages for about five iterations, we show only configurations resulting from an accepted step. The acceptance ratio was in both cases roughly 0.5, which means that we visualize each system for about 5 iterations.



Regular lines represent links with a value of one, dashed and dotted lines represent values two and three. The box is the tail of the worm and the cross is the head for which type I updates are proposed.

4.4.2 Heat Bath Update

A heat bath update can be constructed out of one single step, which is alternated for u and v. We describe the step for u.
First, out of the the $2D$ links neighboring u, one is chosen at random. We call this link $\hat{l}$ and the site which is connected to u through this link is u'. We now need to choose a new value for $k(\hat{l})$. We will construct a heat bath step allowing for the whole range of positive integer linkvalues. Since the field $d(x)$ is constrained to contain only even numbers, we have to move u along the chosen link if the link value is changed by an odd number. We can write the update as

$$k(\hat{l}) \to \tilde{k}(\hat{l}) = k(\hat{l}) + \Delta k, \tag{4.27}$$

$$u \to u' \text{ if } \Delta k \text{ is odd.} \tag{4.28}$$

The heat bath probability to choose $\tilde{k}$ as the new value of $k(\hat{l})$ contains the fields $d(u)$ and $d(u')$ which now depend on $k(\hat{l})$ in a more complicated way, since they must reflect whether u was moved. We define a distribution

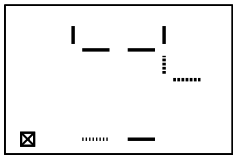
$$\hat{\delta}_i = \begin{cases} 0, & \text{if } i \text{ even} \\ 1, & \text{if } i \text{ odd} \end{cases} \tag{4.29}$$

and use it to write down the heat bath probability to change $k(\hat{l})$ by $\Delta\tilde{k}$:

$$p_{\Delta\tilde{k}(\hat{l})} = \left[\sum_{\Delta k=-k(\hat{l})}^{\infty} \beta^{\Delta k-\Delta\tilde{k}} \frac{(k(\hat{l})+\Delta\tilde{k})!}{(k(\hat{l})+\Delta k)!} \frac{c(d(u)+\Delta k-\hat{\delta}_{\Delta k})}{c(d(u)+\Delta\tilde{k}-\hat{\delta}_{\Delta\tilde{k}})} \frac{c(d(u')+\Delta k+\hat{\delta}_{\Delta k})}{c(d(u')+\Delta\tilde{k}+\hat{\delta}_{\Delta\tilde{k}})} \right]^{-1}. \tag{4.30}$$

This step satisfies balance, and an algorithm constructed of these steps again has the ability do disassemble any given initial configuration, and to construct any given new configuration from the trivial one, hence satisfying ergodicity. Movement over the lattice without changing links can occur (to draw on another animal-metaphor) in a caterpillar-like fashion, where the head moves by changing a bond value by an odd number, and the tail then follows the head by undoing the change of the bond value.

Compared to the probabilities occurring in the Metropolis update, (4.30) is very inconvenient to compute, since it consists of an infinite sum where the terms only start to shrink for $\Delta k > \Delta\tilde{k}$, and the point where truncation is safe is hard to estimate. Also, ratios of c can become large enough to present problems to an algorithmic implementation, as we will see in Section 4.5. In the Ising case however the dependence of (4.30)



on $d(x)$ vanishes (as we will also see in Section 4.5), and the formula reduces to

$$p_{\tilde{k}(\hat{l})} = e^{-\beta}\frac{\beta^{\tilde{k}}}{\tilde{k}!}. \tag{4.31}$$

In this special case, the heat bath update can be employed efficiently, as shown in [13].

4.5 Computing Ratios of c

To implement the algorithm we still need the configuration weights $c(n)$ defined in (4.9). In the new representation of the partition function, the dependence on λ appears only in these values. The integral c satisfies a two-step recursion, that can be obtained by partial integration

$$(n+1)c(n) = 4\lambda c(n+4) + 2(1-2\lambda)c(n+2). \tag{4.32}$$

As we have seen in Section 4.4.1, for a Metropolis type update we only need the ratios of "neighboring" weights c. The ratios needed for a heat bath update can of course be calculated by multiplying the right ratios of neighboring weights:

$$\frac{c(d+\Delta)}{c(d)} = \begin{cases} \prod_{i=1}^{\Delta/2} \frac{1}{\tilde{c}(d+2i)}, & \text{if } \Delta > 0 \\ 1, & \text{if } \Delta = 0 \\ \prod_{i=0}^{-\Delta/2-1} \tilde{c}(d-2i), & \text{if } \Delta < 0 \end{cases}. \tag{4.33}$$

Hence knowing the values of $\tilde{c}$ suffices for both updates. The ratios satisfy the one-step recursion

$$\tilde{c}(n) = \frac{c(n-2)}{c(n)} = \frac{1}{n-1}\left[4\lambda\left(\frac{1}{\tilde{c}(n+2)} - 1\right) + 2\right]. \tag{4.34}$$

The solution for the Gaussian case can already be read off from that recursion. It is

$$\lambda = 0 \to \tilde{c}(n) = \frac{2}{n-1}. \tag{4.35}$$

We can normalize the integral c without changing ratios. If we set $c(0) = 1$, we can write down $c(n)$ for the Gaussian case using (4.33):

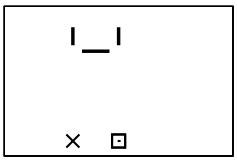
$$c(n) = \frac{(n-1)!!}{2^{n/2}}. \tag{4.36}$$

Rewriting (4.34) as

$$\frac{1}{\tilde{c}(n+2)} = \frac{(n-1)\tilde{c}(n) - 2}{4\lambda} + 1 \tag{4.37}$$

yields the solution for the Ising limit

$$\lambda = \infty \to \tilde{c}(n) \equiv 1 \to c(n) = 1. \tag{4.38}$$



For finite values of λ, the solution can not be read off from the recursion for arbitrary n. However, the recursion can still be used to calculate values of $\tilde{c}$ successively.
Since the representation of numbers in computer memory has limited precision, a recursive step will in general start from a number r_n that differs from the true value r_n^* [18, 22]. We denote this difference $\delta_n = r_n - r_n^*$. One step of the recursion can then be written as

$$r_n^* + \delta_n = f(r_{n\pm1}^* + \delta_{n\pm1}). \tag{4.39}$$

If the derivations δ_n grow as the recursion progresses, the values obtained by use of the recursion lose precision and can in the worst case become worthless after a few steps. On the other hand, if the derivations shrink, precision can be gained, and starting the recursion earlier can even compensate an uncertainty in the starting value. The recursion is then called *stable*.
Writing (4.34) as in (4.39) with a small deviation $\delta_{\tilde{c},n} = \tilde{c}(n) - \tilde{c}^*(n)$ and linearizing in $\delta_{\tilde{c},n}$ yields

$$\delta_{\tilde{c},n} = \underbrace{-\frac{4\lambda}{n-1}\left[\frac{1}{\tilde{c}(n+2)}\right]^2}_{W(\lambda,n)} \delta_{\tilde{c},n+2} + O(\delta_{\tilde{c},n+2}^2). \tag{4.40}$$

The equation describes the propagation of small deviations. It suggests downward stability for $W \leq 1$ and upward stability for $W \geq 1$. However, W depends on λ and n explicitly as well as implicitly through $\tilde{c}$. Therefore, we define a new function

$$r(n) := \sqrt{\frac{2\lambda}{n}} \frac{c(2n)}{c(2(n+1))} = \sqrt{\frac{2\lambda}{n}} \frac{1}{\tilde{c}(2(n+1))} \tag{4.41}$$

which in turn satisfies a recursion. With $\sqrt{2\lambda} = e^{-\gamma}$, we can write

$$\frac{1}{r(n)} = r(n+1) \underbrace{\frac{\sqrt{n(n+1)}}{n+\frac{1}{2}}}_{X(n)} + \underbrace{\frac{2\sinh(\gamma)}{\sqrt{n}(1+\frac{1}{2n})}}_{Y(n,\lambda)} \qquad (n > 0). \tag{4.42}$$

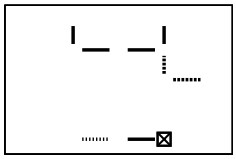
A small deviation will propagate as follows:

$$\delta_{r,n} = -r(n)^2 X(n) \delta_{r,n+1}. \tag{4.43}$$

For convenience, r was "contracted" and shifted in terms of n, so that the recursion (4.42) applies for $n = \{1, 2, 3....\}$ as opposed to (4.32), where the recursion related $\tilde{c}(n)$ with $n = \{2, 4, 6, ...\}$. We have now eliminated the explicit dependence on λ, and since

$$X(n) = 1 - \frac{1}{8}\frac{1}{n^2} + \frac{1}{8}\frac{1}{n^3} + O(n^{-4}), \tag{4.44}$$

we can also neglect the explicit dependence on n for large n.
Then we have $r(n+1) \approx 1/r(n)$. This could be satisfied by a "staggered" solution, distinguishing even and odd values of n. Considering our solutions for the Ising and



Gaussian limits and also remembering that even and odd n correspond to consecutive even values of d, this is implausible, and we therefore make the ansatz

$$r(n) = 1 + \sum_k c_k n^{-k/2}. \tag{4.45}$$

By inserting this ansatz into (4.42), we can calculate an arbitrary number of terms. We find for example

$$r(n) = 1 - \sinh(\gamma)\frac{1}{\sqrt{n}} + \frac{\sinh^2(\gamma)}{2}\frac{1}{n} + \frac{\sinh(\gamma)}{4}\frac{1}{n^{3/2}} + \frac{1-2\sinh^4(\gamma)}{16}\frac{1}{n^2} + O\left(\frac{1}{n^{5/2}}\right). \tag{4.46}$$

For $\sinh(\gamma) = 0 \leftrightarrow \lambda = 0.5$ we have

$$r(n) = 1 + \frac{1}{16}\frac{1}{n^2} + O(n^{-4}). \tag{4.47}$$

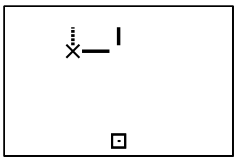
Inserting this into (4.43), we find that the terms of order n^{-2} in r^2 and X cancel, and we are left with

$$r(n)^2 X(n)|_{\lambda=0.5} = 1 + \frac{1}{8}\frac{1}{n^3} + O(n^{-4}). \tag{4.48}$$

At $\lambda = 0.5$, we would therefore preferably use the upward iteration. However, the deviation stays almost constant either way. If we were for example to start from $r(100)$ and iterate all the way down to $r(1)$, a small deviation would grow by a factor of roughly 1.15.
If we move away from $\lambda = 0.5$ at fixed n, the order $n^{-1/2}$ term in r will become dominant, and we have $r^2 > 1$ for $\lambda < 0.5$ and $r^2 < 1$ for $\lambda > 0.5$, clearly indicating the stable direction of recursion. After these considerations, we expect the following strategy to be successfull and efficient for calculating values of $\tilde{c}(n)$ for $n = \{2, 4, ..., n_{max}\}$ at a given λ:

For $\lambda < 0.5$

- Calculate $r(n_{max}/2 - 1)$ by numerically solving the integrals $c(n_{max}), c(n_{max} + 2)$ and using (4.41), or by using enough terms of the series expansion to achieve the desired precision.
  One strategy is to calculate $r(n)$ for a high n, where the expansion is very accurate even with only a few terms, and then iterate down unto $r(n_{max})$, again gaining precision.
- Use the downward iteration to calculate $r(n)$ down unto $r(1)$.
- Use (4.41) to calculate $\tilde{c}(n)$ down unto $\tilde{c}(4)$ from r.
- Calculate $\tilde{c}(2)$ numerically.



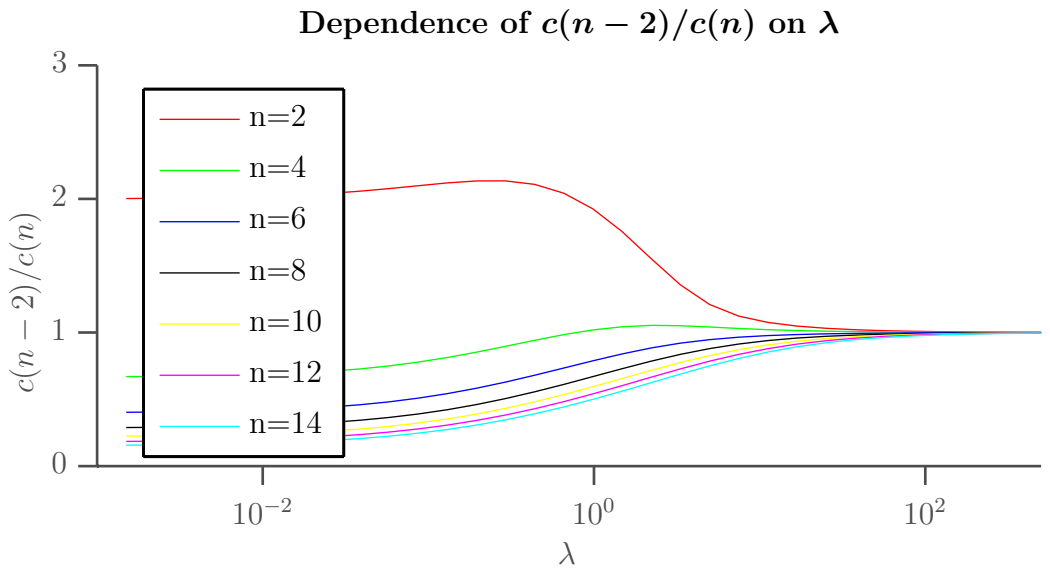


Figure 4.3: The plot shows the values of neighboring ratios of configuration weights c. The curve progression is not monotonous for the lowest values of n.

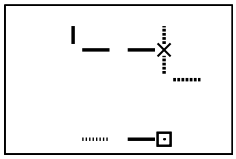
For $\lambda \geq 0.5$

- Calculate $\tilde{c}(2)$ and $\tilde{c}(4)$ numerically.
- Calculate $r(1)$ using (4.41) to start the recursion and iterate up to receive all desired $r(n)$.
- Use (4.41) to calculate $\tilde{c}(n)$ up until $\tilde{c}(n_{max})$ from r.

For verification, this method was used to compute values of $\tilde{c}(n)$ for the lowest n and a range of λ between 10^{-3} and 10^{3}. Values were checked against a numerical integrator from the GNU Scientific Library [28], and against the integration routine of Wolfram Mathematica [29] with an increased precision of 20 places. The highest deviations found were $O(10^{-15})$. Figure 4.3 shows the found values and gives an idea of how all $\tilde{c}(n)$ converge towards the analytically known limits.

4.6 Observables

In this chapter, the extraction of observables from the reformulated field theory will be described. The averaging of an observable A over the link ensemble, denoted with $\langle\langle A \rangle\rangle$, was defined in (4.18).



4.6.1 Greens Functions

Using translation invariance in (4.21) to average over the whole volume, we can write general Greens functions as

$$G(z) = \langle \varphi(0)\varphi(z) \rangle = \frac{\langle\langle \delta_{v-u,z} \rangle\rangle}{\langle\langle \tilde{c}(d(u))\delta_{u,v} \rangle\rangle}. \tag{4.49}$$

We notice, that in order to compute ratios of Greens functions, only a histogram of all distances $u - v$ occurring during the simulation is necessary. The accumulation of this histogram has a cost of one operation per update. The normalizing denominator depends not only on the movement of u and v but also on the values of the surrounding links. In the spirit that memory is cheaper than computing time, it has been proven beneficial to keep the field $d(x)$ as an auxiliary field and update it after each step, even though it is completely defined by the link field and the field insertions.

4.6.2 Two-Point Susceptibility

The two-point susceptibility has a particularly simple estimator,

$$\chi_2 = \sum_z G(z) = \frac{1}{\langle\langle \tilde{c}(d(u))\delta_{u,v} \rangle\rangle}, \tag{4.50}$$

since the sum over all possible delta-functions will always yield exactly one.

4.6.3 Two-Point Functions in Momentum Space

We can derive the two-point function in momentum space from (4.49) as follows:

$$\tilde{G}(p) = \sum_x e^{ipx} G(x) = \frac{\langle\langle \cos(p(u-v)) \rangle\rangle}{\langle\langle \delta_{u,v}\tilde{c}(d(u)) \rangle\rangle}. \tag{4.51}$$

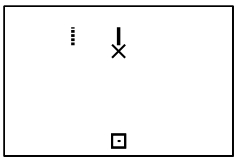
Again, it has proven effective to create a table containing the values of the cosine for all lattice distances before starting the simulation.

4.6.4 Second Moment Mass

The second moment mass, as defined in (2.50), can be extrated using (4.51)

$$\frac{m^2}{m^2 + \hat{p}_*^2} = \langle\langle \cos(p_*(u-v)) \rangle\rangle . \tag{4.52}$$

In the hypercubic geometries that we will consider, statistics can be enhanced by averaging over p_* in all dimensions.



4.6.5 Energy

The definition of the energy as the sum over nearest neighbor correlations can be translated directly with (4.49),

$$E = \frac{1}{D}\sum_{\mu} G(\mu) = \frac{1}{2D}\frac{\left\langle\left\langle \delta_{|u-v|,1}\right\rangle\right\rangle}{\langle\langle \tilde{c}(d(u))\delta_{u,v}\rangle\rangle}. \tag{4.53}$$

Another estimator for the energy can be derived by computing the derivative of the partition function's logarithm with respect to β. On the one hand, we can write the energy as

$$\frac{1}{N_l}\frac{\partial}{\partial\beta}\ln Z(\emptyset) = \frac{1}{N_l}\sum_{l=<xy>}\langle\varphi(x)\varphi(y)\rangle = E. \tag{4.54}$$

The right hand side is simply (4.53) after averaging over the volume. Here, we have written the partition function in terms of the field, as in (4.2). If on the other hand we write it in terms of links, as in (4.8), we find

$$E = \frac{1}{\beta}\frac{1}{N_l}\frac{\langle\langle \tilde{c}(d(u))\delta_{u,v}\sum_l k(l)\rangle\rangle}{\langle\langle \tilde{c}(d(u))\delta_{u,v}\rangle\rangle}. \tag{4.55}$$

4.6.6 Ising Energy

In the Ising case, equation (4.55) simplifies to

$$E_{Ising} = \frac{1}{\beta}\frac{1}{N_l}\frac{\langle\langle \delta_{u,v}\sum_l k(l)\rangle\rangle}{\langle\langle \delta_{u,v}\rangle\rangle} \quad \text{for} \quad \lambda = \infty. \tag{4.56}$$

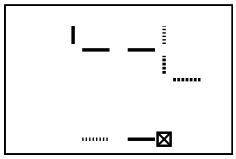
For finite λ this remains a meaningful quantity, since it can be expressed in terms of correlation functions. Again this is achieved by first writing the estimator in terms of the partition function, and then evaluating the gained expression with the partition function written in terms of the fields,

$$\left\langle\left\langle \delta_{u,v}\sum_l k(l)\right\rangle\right\rangle = L^D\frac{Z_0}{\mathcal{Z}}\frac{\partial}{\partial\beta}\ln Z_0 = \frac{Z_0}{\mathcal{Z}}\left\langle\varphi^2(0)\sum_{l=<xy>}\varphi(x)\varphi(y)\right\rangle. \tag{4.57}$$

The right hand side can be expressed as a sum of two-point functions and we finally find

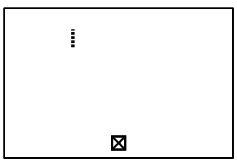
$$\frac{\langle\langle \delta_{u,v}\sum_l k(l)\rangle\rangle}{\langle\langle \delta_{u,v}\rangle\rangle} = \sum_{\mu}\left[L^D G(\mu) + \frac{2}{G(0)}\sum_x G(x)G(x+\mu)\right]. \tag{4.58}$$

For $\lambda = 0$, this can be expressed in terms of Fourier sums, as shown in (2.44).



4.6.7 Connected Four-Point Susceptibility

The connected four-point susceptibility $\chi_{4,c}$ was defined in (2.55). It contains a sum over four point functions which is at first glance hard to access with the worm algorithm as described here, since it was explicitly tailored to the calculation of two-point functions by choice of the enlarged ensemble (4.17). In Chapter 7, we will present and discuss a procedure which allows the extraction of the renormalized coupling from two two-insertion ensembles. This procedure held the potential to allow for an efficient calculation of g_R and was therefore examined in some detail.
The choice of the enlarged ensemble made in this chapter is however not the only possible one. Sums over partition functions with four field insertions is equally possible. In Appendix A.1, we present some thoughts on the extraction of four-point functions from such an ensemble.



5 Testing the Implementation

After all prerequisites for an implementation are met, we have proceeded by writing a C implementation of the algorithm. We have preferred the Metropolis type update step over the heat bath step for its much easier acceptance probabilities. The implementation uses the analytically known weigths $\tilde{c}$ in the corresponding limits and calculates them recursively anywhere else. Random numbers were generated with ranlux, luxury level one [30]. Extensive testing of the simulation was performed in the Ising and Gaussian limits as well as for finite values of λ. In the following we will proceed with the presentation of data obtained during these runs.

5.1 Testing in the Ising Limit

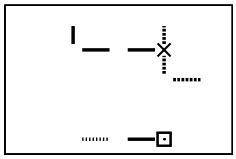
We tested at the critical points in two dimensions $\beta = \ln(1+\sqrt{2})/2$, and in three dimensions $\beta = 0.22165$ [20].

D	L	$L^{\gamma/\nu}/\chi$	E (4.53)	E (4.55)	mL
2	16	0.9186(22)	0.72617(69)	0.72641(36)	1.1026(53)
2	32	0.9184(21)	0.71595(60)	0.71653(23)	1.1051(57)
2	64	0.9157(24)	0.71141(54)	0.71205(14)	1.1008(64)
2	128	0.9142(32)	0.70921(49)	0.709517(85)	1.0965(73)
2	256	0.9155(33)	0.70808(45)	0.708345(52)	1.1013(82)
3	16	0.6606(17)	0.34466(25)	0.34478(14)	1.5610(31)
3	24	0.6572(18)	0.33811(18)	0.338477(84)	1.5644(32)
3	32	0.6560(19)	0.33557(16)	0.335600(61)	1.5626(32)
3	48	0.6544(20)	0.33320(12)	0.333235(38)	1.5613(34)
3	64	0.6596(23)	0.33223(10)	0.332154(25)	1.5761(34)

Table 5.1: Results of simulations of the Ising model in two and three dimensions. We show values for susceptibility, second moment mass and results for two different estimators of the energy.

Results are shown in Table 5.1. We have presented the susceptibility, the energy using next-neighbor and link population estimators as well as the second moment mass. Energy and susceptibility are consistent within errors with [20]. The susceptiblity diverges at the critical point according to the power law

$$\chi_2 = L^{\gamma/\nu}, \tag{5.1}$$



which we have used to scale our data. In two dimensions $\gamma/\nu = 7/4$ is known exactly. In three dimensions we have used the numeric result $\gamma/\nu \approx 1.963$ [31]. Masses were measured as well, and we find correlations on the scale of the lattice for every lattice size.

Each run consisted of four independent replica, each measuring for 10^6 iterations after 10^4 iterations of thermalisation. The longest measurement of a single replica consisted of $64^3 \cdot 10^6 \approx 2 \cdot 10^{11}$ updates and took 10.2 hours of CPU time on one core of an AMD C2660 processor.

Errors show virtually no dependence on the lattice size. As pointed out in [13], this signifies the absence of critical slowing down. The dynamic behavior of the worm simulations of the Ising model has been examined extensively in two and three dimensions in [9]. Results for four and five dimensions were presented in [10]. We consider the well-behavedness of the algorithm in the Ising model as established.

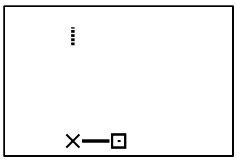
5.2 Testing in the Gaussian Limit

In Section 2.5, we have discussed the calculation of Greens functions of the free theory from discrete sums. We used these formulas to check the results of all simulations of the free theory and found consistency within errors. To give a first impression of achieved precisions, we compile some sample data obtained from a single simulation of 10^6 iterations in Table 5.2. The simulation consisted of $16^3 10^6 \approx 4 \cdot 10^9$ updates and

Observable	Estimate	Exact value	Δ/σ
χ_2	817.(75.)	768.500	0.65
$G(0)$	0.91(1)	0.902	0.63
$G(1)$	0.41(1)	0.402	0.63
$G(2)$	0.28(1)	0.273	0.64
mL	0.97(4)	1.000	-0.65
E	0.41(1)	0.402	0.63
E_I	0.50(3)	0.481	0.60

Table 5.2: Results of a simulation of the Gauss model on a 16^3 lattice with $m_0 L = 1$, accumulated during 10^6 iterations. We compare all observables with the exact values, which we have calculated as pointed out in Section 2.5. In the last column the deviation from the exact value in units of the error is given.

took roughly 9 minutes of computing time on one core of an AMD C2660 processor. This is comparable to the time per update measured in the Ising case. All results agree within errors with the exact values.



5.3 Testing at finite λ

In order to test the implementation and to get a first impression of its performance, we have measured lines of equal criticality in the symmetric phase on a 16^3 lattice. This involved tuning the parameter β in order to achieve for a given coupling strength the sought for criticality, wich we measured by comparing correlation length and lattice size. We have chosen to measure one line very close to criticality with $mL = 1$, and one with a shorter correlation length by tuning for $mL = 4$. The data was verified with standard metropolis simulation of φ^4 theory discussed in Section 3.3.3.

5.3.1 Critical Line in Three Dimensions

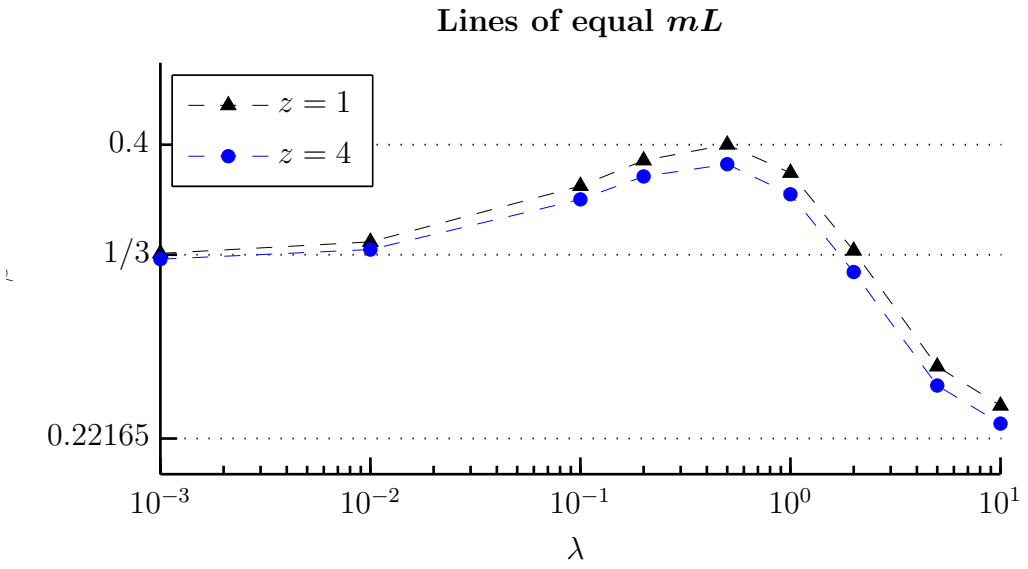
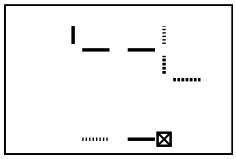


Figure 5.1: Lines of equal mL in three dimensions. The dashed lines only connect the data points.

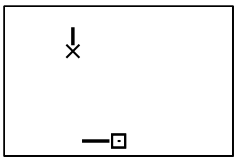
The results of the tuning for several values of λ are presented in figure 5.1. Perturbation theory tells us that the line has a positive slope at $\lambda = 0$ [11]. The critical point of the three dimensional Ising model is located approximately at $\beta = 0.22165$, considerably lower than the critical point at $\lambda = 0$. The non-monotonous curve progression is therefore expected. The lines reach their maximum around $\lambda = 1/2$, which has also been found to be the case in four dimensions [14]. We expect to find systems very close to criticality by setting $mL = 1$. The convergence of the measured line towards the known critical points in the limits supports this expectation. After tuning, measurements at the found parameter values were conducted with the standard Metropolis as well as the worm algorithm. All measurements consisted of 10^6 sweeps or iterations respectively. In Table 5.3 some values are presented for comparison and the full data is listed in Appendix B.1. The results are compatible with each other as well as with the tuning target.



λ	β	mL		$G(0)$	
		Worm	Metropolis	Worm	Metropolis
0.5	0.4002	0.9989(64)	1.010(10)	0.63490(35)	0.63479(37)
1.0	0.3830	1.0067(66)	0.994(11)	0.64289(24)	0.64281(26)
2.0	0.3358	0.9984(62)	0.993(13)	0.70717(13)	0.70715(19)
5.0	0.2654	0.9904(62)	0.995(18)	0.860766(48)	0.860804(90)
0.5	0.3883	3.9953(76)	4.036(49)	0.585569(77)	0.585605(65)
1.0	0.3700	4.0104(70)	3.949(48)	0.606664(55)	0.606696(51)
2.0	0.3228	3.9906(64)	3.992(54)	0.683603(37)	0.683647(35)
5.0	0.2538	4.0049(64)	4.182(85)	0.851687(14)	0.851674(22)

Table 5.3: Comparison between worm and standard Metropolis simulation. We compare calculated masses as well as $G(0) = \langle \phi^2 \rangle$. The upper half shows measurements along the line $mL = 1$, the lower half along the line $mL = 4$.

A quick comparison of the errors suggests that the performance of the worm algorithm depends on observable in question. While the mass is always estimated more precisely by the worm algorithm, errors of $G(0)$ are comparable between both simulations. Also, the standard metropolis simulation performs worse for the more strongly interacting theory, while the opposite is the case for the worm algorithm.



6 Dynamical Critical Behavior

After giving a first impression of the algorithm's performance in the last chapter, we will now proceed with a systematic analysis of its critical behavior. We will use the ultraviolet finite size renormalization scheme that was outlined in Section 2.8. We measure autocorrelation times during two approaches of the continuum limit. In the first set of measurements, we choose $mL = 4$. By doing so, we make sure that our lattice is larger than the correlation length and finite size effects are small. In the second set, we will set $mL = 1$. With correlation length equal to the lattice size, we expect to achieve a higher proximity to the continuum limit.
We have examined φ^4 theory in two, three and four dimensions. We present results for the Gauss model and for the interacting theory close to the maximum of the critical line at $\lambda = 0.5$. Intuitively, we expect a monotonous interpolation of the dynamical exponents towards the Ising limit. The measurements in the interacting theory will help to verify or falsify this expectation.
Except in one case, we have conducted four independent measurements, each with a run length of 10^6 iterations. This has been sufficient to precisely estimate the autocorrelation of all physical observables.

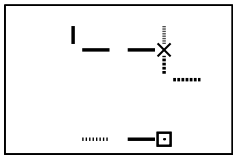
6.1 Examined Observables

We have examined four characteristic observables. The energy E contains information on the decorrelation of the global link field and is therefore of systematic interest. The second moment mass m is a low-momentum observable and depends on the distance histogram for all distances. The estimator for the two-point susceptibility χ_2 is at the same time the denominator for the estimation of all two-point functions. It also depends on $\tilde{c}(d)$, which in turn depends on the "local" link field, i. e., the links surrounding sites where head and tail of the worm meet. We have also included the two-point function of distance zero $G(0)$ in our plots. Since all two-point functions have shown similar dynamical properties, we will use only $G(0)$ to exemplify them.

6.2 Critical Exponents

We have committed to a preset mode of fitting critical exponents. Close to the critical point, the scaling laws are in leading order power laws in L [23],

$$\tau_A = \alpha_A L^{z_A}. \tag{6.1}$$



The exponent z_A is the critical exponent of the observable A. In our fits, we exclude all points were the power law is not yet the dominant contribution to the autocorrelation. We decide whether or not this is the case based on the reduced χ^2 value of our fits. We accept only fits with a reduced χ^2 smaller than two.
The scaling functions that we present in this chapter are primarily meant to describe the dynamic behavior in the parameter space examined in this thesis. They are also expected to allow far an interpolation towards the critical point in reasonable limits. We would however like to stress here, that they are not meant as exact predictions of autocorrelation times at the critical point.

6.3 Critical Behavior in the Gauss Model

We will begin with the critical behavior in the Gauss Model, starting with the case of two dimensions.

6.3.1 Two Dimensions

The measured autocorrelation times are represented in Figure 6.1. The corresponding data can be found in Appendix B.2.1.

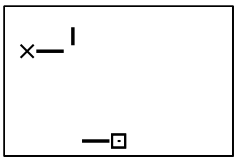
The first set of measurements was taken at $mL = 4$. Here, we find autocorrelation times of up to 300 iterations length. Except for the two-point function, all observables exhibit very similar autocorrelations and the critical exponents are all consistent with $z = 1.6$. This exponent is lower than that of the standard Metropolis simulation presented in Section 3.3.3, but still signifies substantial slowing down.

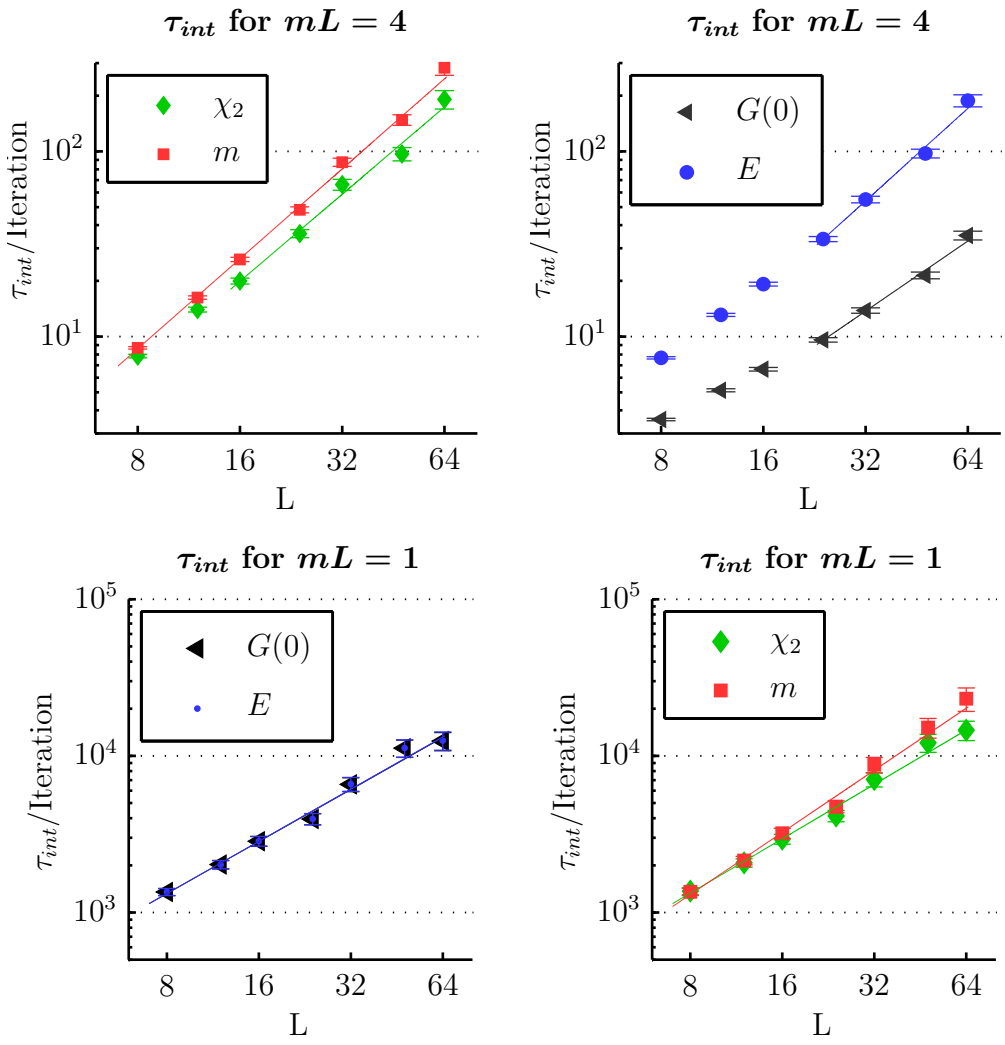
The two point function shows lower autocorrelations already at the smallest lattice, and we find a critical exponent of $z \approx 1.2$. To investigate this behavior we show in Figure 6.2 again $G(0)$ together with its primary estimators. Both estimators show integrated autocorrelation similar to those of energy and mass. However, these correlations cancel in the two-point function. As a result, $G(0)$ shows autocorrelation times of one order of magnitude smaller than those of all other observables. The lower critical exponent of $G(0)$ suggest that the cancellations grow larger towards the continuum limit. We find the same behavior for $G(1)$ and $G(2)$.

In the second set of measurements, we started closer to the critical point. As expected, we measured longer autocorrelation times than at $mL = 4$. They now reached a length of order 10^4 iterations forcing us to run longer simulations in order to obtain precise estimates. For each lattice size we have conducted four independent measurements, each consisting of $5 \cdot 10^6$ iterations.

We find very similar behavior for all observables. In fact, $G(0)$ and E have almost identical autocorrelation times.

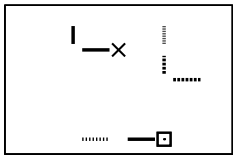
Closer to the critical point, our results are better approximated by the power law which is reflected by lower χ^2 values of all our fits. In the last series of measurements, m has a slightly higher critical exponent. Remarkably, although autocorrelation times are higher for $mL = 1$, we measure lower critical exponents which is unusual dynamic behavior.





mL	Observable	α	z	χ^2/dof
4	χ_2	0.26(22)	1.563(59)	1.78
4	m	0.300(55)	1.613(20)	1.40
4	E	0.17(29)	1.657(66)	1.40
4	$G(0)$	0.17(22)	1.256(52)	1.37
1	χ_2	120.85(15)	1.153(48)	0.77
1	m	84.09(16)	1.317(52)	0.92
1	E	132.13(14)	1.107(46)	0.78
1	$G(0)$	133.00(14)	1.104(46)	0.78

Figure 6.1: Critical dynamical behavior in the two-dimensional Gaussian model for four observables. Fit parameters are listed in the table. Lines connect points that were used for fitting.



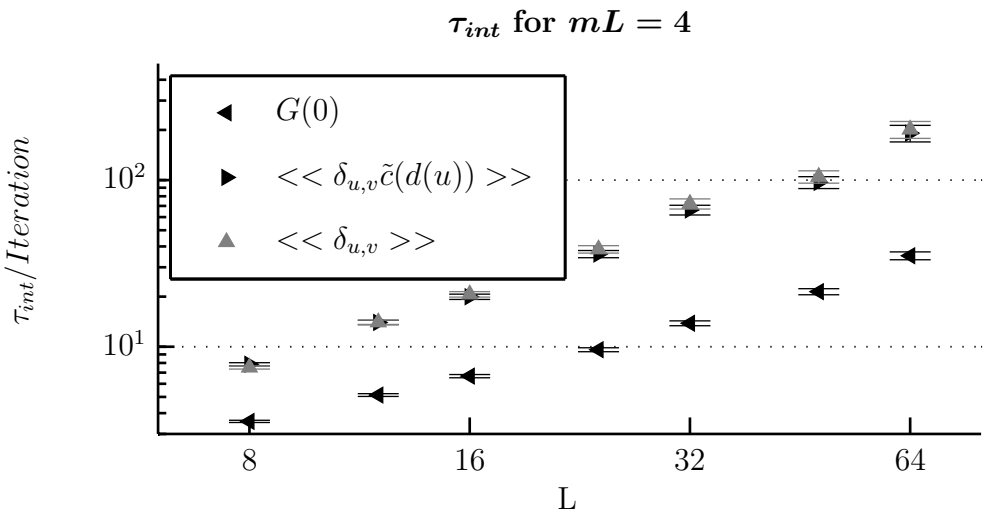


Figure 6.2: Cancellation of autocorrelations of the two primary estimators of $G(0)$. Measurements taken with $D = 2$, $\lambda = 0$, $mL = 4$.

6.3.2 Three Dimensions

Next we present our results for the Gauss Model in three dimensions. Data is shown in Figure 6.3 together with fit parameters.

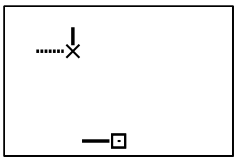
We find much more moderate autocorrelations than in the previous case of two dimensions. Critical exponents for both sets are all slightly smaller than one except for the two-point function, which again shows exceptional behavior. In the measurement with $mL = 1$, autocorrelation times of $G(0)$ show a power law behavior for the three smallest lattice sizes, and then remain almost constant. This is again caused by cancellations, as observed in the two dimensional case. For $mL = 4$ the autocorrelations of $G(0)$ are almost constant on all lattices. Susceptibility, mass and energy again show very similar behavior.

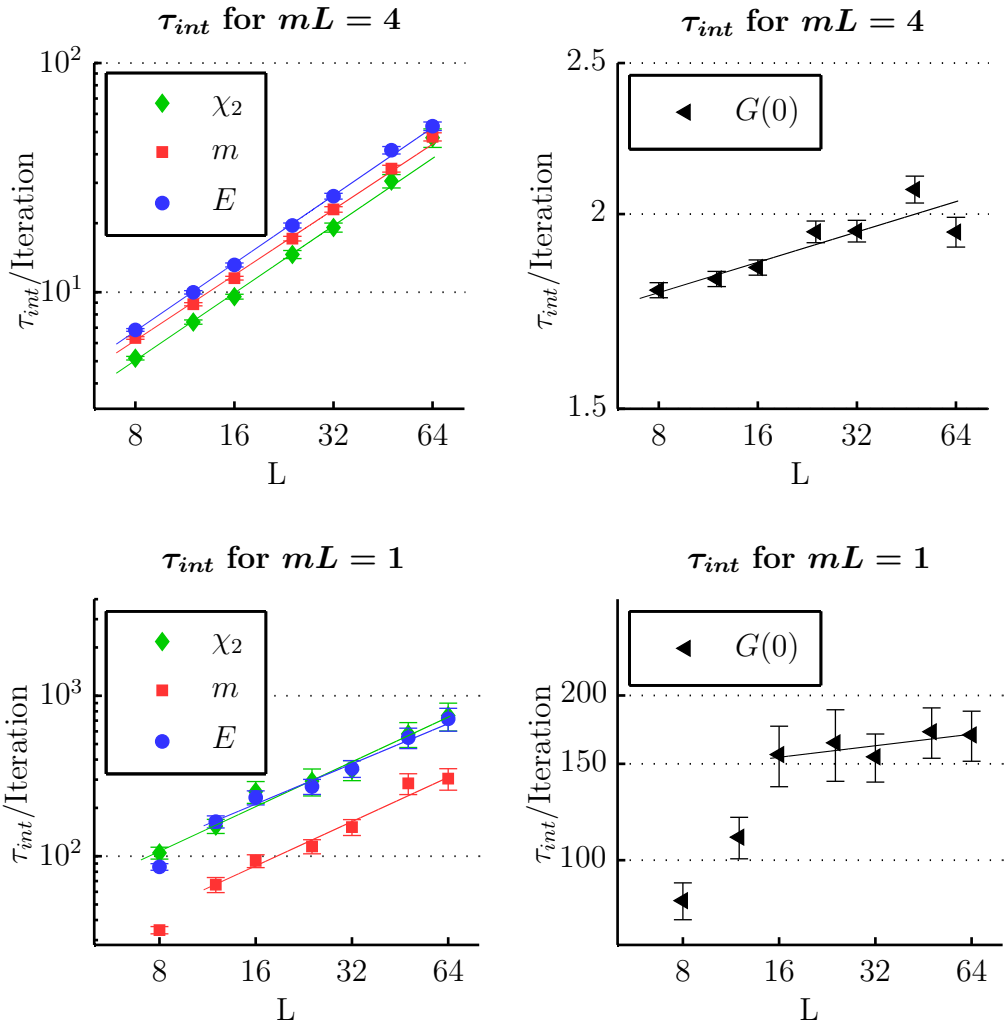
Previous measurements of integrated autocorrelation times of the energy on lattices of up to 64^3 sites have been published in [8]. A logarithmic dependence of the integrated autocorrelation times on the lattice extent was documented, which is not consistent with our results. Since in [8] the mass parameters used for the simulation are not given, we are unable not make a quantitative comparison.

6.3.3 Four Dimensions

We finally examine the physical case of four dimensions. In order to keep our computational effort within reasonable limits, we limited our measurements to smaller lattice extents. The largest lattice consisted of 24^4 sites. The fits are shown in Figure 6.4 and the data can be found in Appendix B.2.1.

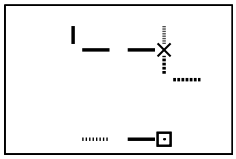
Again, we measure much shorter autocorrelation times than in the previous case. All





mL	Observable	α	z	χ^2/dof
4	χ_2	0.663(60)	0.975(21)	1.63
4	m	0.747(58)	0.990(18)	0.44
4	E	0.875(38)	0.983(13)	0.80
4	$G(0)$	1.556(25)	0.0646(84)	1.64
1	χ_2	15.61(25)	0.926(72)	0.45
1	m	6.85(37)	0.916(87)	0.67
1	E	21.03(34)	0.832(84)	0.42
1	$G(0)$	126.94(56)	0.07(10)	0.10

Figure 6.3: Critical dynamical behavior in the three-dimensional Gaussian model for four observables. Fit parameters are listed in the table. Lines connect points that were used for fitting.



critical exponents are well below one. The two point function now even exhibits critical speeding up. At $mL = 1$, this effect is faster than can be described by a power law, which is why we have not attempted to fit the curve progression. At $mL = 4$, we were also unable to fit the curve progression to a power law, but the plot suggests a tendency towards completely uncorrelated configurations.

The observables now show more of a distinct behavior. Particularly at $mL = 4$, the energy shows longer autocorrelation times and a higher critical exponent than χ_2 and m. This coincides with the smallest autocorrelation times measured for these three observables so far.

6.3.4 Discussion

We have presented our data for the Gaussian case. Comparing the results for each dimension with each other and with the analysis presented in [9], we record the following observations:

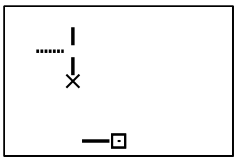
- In contrast to the Ising case, significant critical slowing down is present in the two- and three-dimensional Gaussian model.
- In the cases considered here, the occurring autocorrelations are shorter for higher dimensions. If we compare measurements $mL = 1$, $L = 24$, we find autocorrelations are one order of magnitude smaller from dimension to dimension, starting with $\tau_{int} = O(10^4)$ in two dimensions.
- In two and three dimensions, E, χ_2 and m show almost identical critical behavior. This is also the case in four dimensions in the set of measurements closer to criticality.

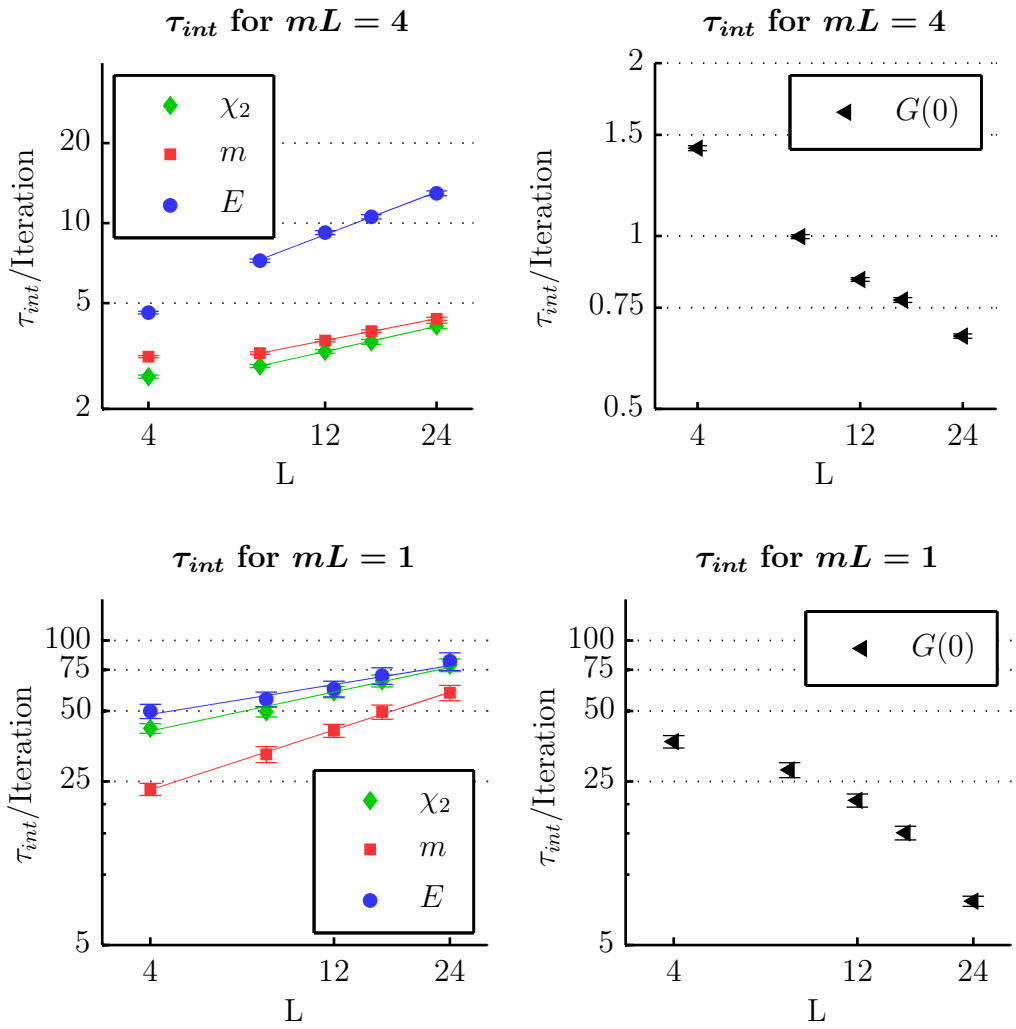
In this discussion, we will present some heuristics in terms of which the observed dynamical behavior is, in our opinion, plausible.
As pointed out in Section 3.4, the slowest mode of relaxation τ_{exp} is of special interest to the dynamical behavior of a system. In [9] the observable

$$\mathcal{N} = \left\langle \left\langle \sum_l k(l) \right\rangle \right\rangle \tag{6.2}$$

was identified as the slowest mode of the system in the two- and three-dimensional Ising case. We have collected data for this observable in all our runs, and values of $\mathcal{N}/N_l$ are listed in Appendix B.2. The observable can be related to a rather complicated sum of products of Greens functions by repeating the calculation done for the Ising Energy (4.56) for all distances.

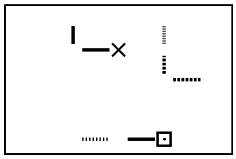
Since all two-point functions can be calculated from the distance histograms and the susceptibility, the observable $\mathcal{N}$ is of no particular physical interest, except in closed path configurations, in which it appears in the estimator for the energy (4.54). However, we will make the case that the system's long-timescale behavior in the Gaussian model is dominated by fluctuations of the link population.





mL	Observable	α	z	χ^2/dof
4	χ_2	1.51(0.05)	0.31(2)	0.05
4	m	1.84(0.03)	0.27(1)	0.02
4	E	2.40(0.05)	0.53(2)	0.60
1	χ_2	25.287(99)	0.34(3)	0.37
1	m	10.3(3)	0.55(9)	0.05
1	E	27.0(3)	0.34(9)	0.04

Figure 6.4: Critical dynamical behavior in the four-dimensional Gaussian model for four observables. Fit parameters are listed in the table. Lines connect points that were used for fitting.



In the Ising case, transition probabilities are independent of the "local link field" $d(x)$. The movement of the insertions can be considered to be only weakly connected to the values of links. In the heat bath probabilities (4.31), this is particularly visible, since they are also independent of the link to be changed. After the new link value has been selected, the old value plays a rôle in deciding whether the insertion is moved across the link. However, it is only of consequence if the change of the link is even or odd. There is no direct dependence on the absolute initial value of the link.

For finite interaction strengths λ, the local link field gains influence as the function $\tilde{c}(d)$ changes as depicted in Figure 4.3. We also expect the local link field to influence transition probabilities more strongly, if it shows a higher population.

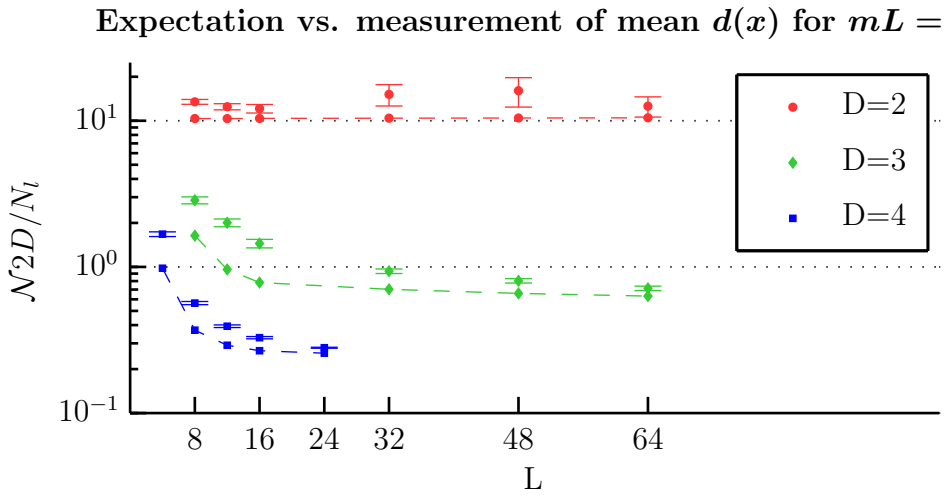


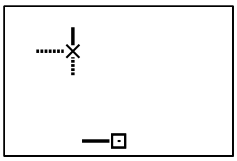
Figure 6.5: Comparison of the right hand side of (6.4) with the measurements taken in two, three and four dimensions for $mL = 1$. Measured points have errorbars, points representing our expectation are connected with broken lines.

It is possible to develop an expectation of the mean link population in the Gaussian case by assuming perfect equilibration of the link field in respect to u and v, similar to the argumentation in [9]. Under this assumption, we make the following approximation using the Ising Energy defined in (4.56):

$$\left\langle \left\langle \delta_{u,v} \sum_l k(l) \right\rangle \right\rangle = \beta N_l \left\langle \langle \delta_{u,v} \rangle \right\rangle E_I \tag{6.3}$$

$$\mathcal{N} \approx \beta N_l E_I \tag{6.4}$$

To check the validity of this approximation, we compare it to the measurements of $\mathcal{N}$ in Figure 6.5. Since we are interested in the mean value of $d(x)$, we compare $\frac{2D}{N_l}\mathcal{N}$ with $2\beta D E_{Ising}$. The comparison suggests, that our approximation yields a lower bound for



$\mathcal{N}$ but can also be considered an order of magnitude statement.

In respect to our original argument, we see that similarly critical systems have very different mean link values, depending on their dimension. This is directly connected to the systems Energy, which is of course independent of the algorithm.

It is our hypothesis that compared to the Ising case, the different form of $\tilde{c}(d)$ and the stronger population of the links in the Gaussian case lead to a stronger connection between the link values and the movement of the worm over the lattice. This results in a higher overlap of the relaxation modes of autocorrelations of "histogram observables" like $\langle\langle\delta_{u,v}\rangle\rangle$ with the observable $\mathcal{N}$. In particular, we expect a higher overlap in lower dimensions.

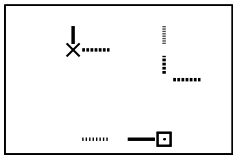
In order to further investigate this hypothesis, we have conducted a correlation analysis between the observable $\mathcal{N}$ and $\langle\langle\delta_{u,v}\rangle\rangle$ along the critical line measured in Section 5.3.1 for the three-dimensional theory. We have no expectations of the functional relation, and therefore use the non-paremetric Spearman Rank-order coefficient as a measure of correlation. The Spearman coefficient r_s of two observables α and β is defined by [22]

$$r_s(\alpha,\beta) = \frac{\sum_i \left(b_\alpha^i - \bar{b}_\alpha\right)\left(b_\beta^i - \bar{b}_\beta\right)}{\sqrt{\sum_i \left(b_\alpha^i - \bar{b}_\alpha\right)}\sqrt{\sum_i \left(b_\alpha^i - \bar{b}_\alpha\right)}}. \tag{6.5}$$

where b_α^i is the *rank* of a_α^i and $\bar{b}_\alpha^i$ is its mean. The rank is the position of an estimate a_α^i in an ordered list of all estimates. The correlation coefficient is normalized and takes values between -1 and 1. Coefficients close to the extreme values indicate strong correlation or anticorrelation respectively, while values close to zero indicate uncorrelated observables. We have measured the correlation coefficient on three different timescales $\mathcal{T}$ by averaging over $\mathcal{T}$ iterations before calculating r_s. The measurements consist of $5 \cdot 10^5$ iterations. The longest timescale considered is $\mathcal{T} = 100$ which leaves $5 \cdot 10^3$ bins. We expect this to allow for a meaningful correlation analysis. The results are presented in the first plot of Figure 6.6.

On the scale of one iteration, we see a small anticorrelation and virtually no dependence on λ. After averaging over 10 iterations, we find a stronger anticorrelation, which is almost constant for higher values of λ. In the area of small λ, correlations grow drastically. The same behavior is found on the largest time scale of 100 iterations with again stronger correlations. This indicates an independent short-time behavior of $\langle\langle\delta_u, v\rangle\rangle$, and a behavior on long time scales correlated with $\mathcal{N}$ more strongly for smaller λ. The results appear to be consistent with our hypothesis.

The assumed growing overlap coincides with rising autocorrelations of $\mathcal{N}$ towards the Gaussian limit. Additionally to our measurements of $\mathcal{N}$ in the Gaussian limit, we have measured autocorrelation times of $\mathcal{N}$ along the by now well known critical line in three dimensions. The results are shown in the second plot of Figure 6.6.
The integrated autocorrelation times of $\mathcal{N}$ remain moderate for $\lambda \geq 5$ but grow very fast for $\lambda \leq 1$ and reach $O(10^3)$ in the Gaussian limit. According to our previous correlation analysis, we expect autocorrelations on this long time scale, to carry over to histogram



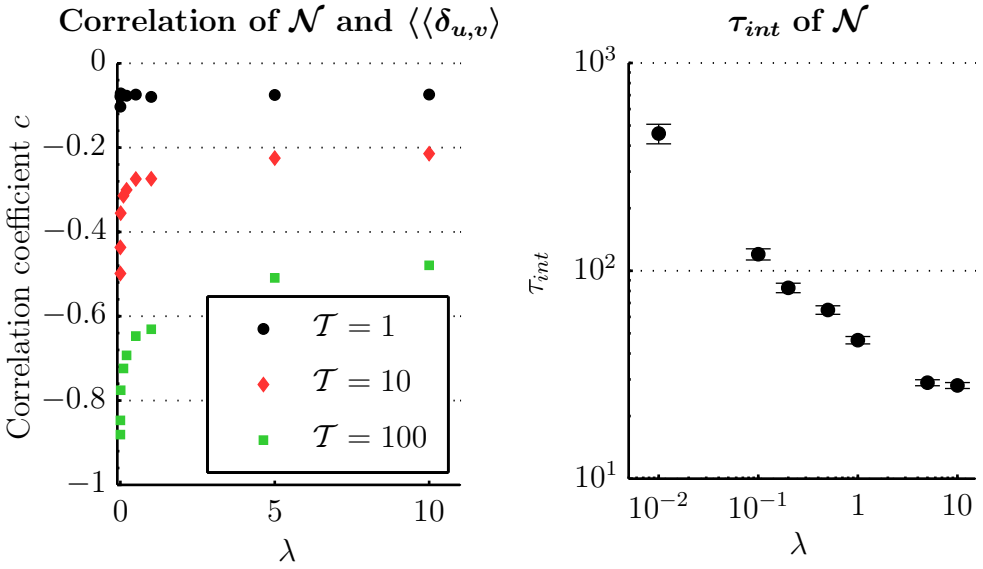
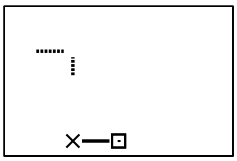


Figure 6.6: Results of simulations on lattices of 16^3 sites at $mL = 1$. Left: Correlation between $\mathcal{N}$ and $\langle\langle\delta_{u,v}\rangle\rangle$ according to (6.5) over λ on different timescales $\mathcal{T}$. Right: Logarithmic plot of integrated autocorrelation times of $\mathcal{N}$ for $L = 16, D = 3, mL = 1$ and several values of λ.

observables.

In our measurements in the Gaussian limit, we have found that $\mathcal{N}$ shows the longest autocorrelations for every set of paremeters. We find the lowest values in four dimensions for $mL = 4$ and also the lowest relative values compared to the autocorrelation of the mass, which we consider a histogram observable. This is also the case showing the lowest mean link values, and the only case, where observables show distinct behavior. We suspect the overlap in this case to be small enough to allow histogram observables to "decouple" from the link field, similar to the Ising limit. If we find shorter autocorrelation times and lower link populations in the interacting case, which we expect in terms of an interpolation towards the Ising case, we should continue to find observables showing distinct behavior. Since the correlation between the link field and histogram observables quickly becomes weaker as λ grows, we also expect to find strongly reduced critical slowing down already at small interaction strenghts.

If the heuristics assembled in this discussion are sustainable, the algorithm's performance could be significantly enhanced by introducing a mechanism capable of rapidly decorrelating the link field. However, we so far see no promising approach to achieve this. It is possible to introduce a third step which exclusively changes the link field at possibly large scales. One possibility is to use a Metropolis-type step [9], restricted to adding or subtracting closed paths. However, the effort to calculate acceptance probabilities of a step proposing to change n_c links would be comparable to the effort of n_c type I steps. We have experimented with a third step adding or subtracting the elementary



closed path, i. e. a link with value two, to the value of a randomly chosen link but were not successful in reducing the autocorrelation of $\mathcal{N}$. The detailed description of the new step and results can be found in Appendix A.2.

The results obtained so far suggest a more favorable dynamical behavior for larger values of λ. Ultimately, we are interested in the interacting theory far away from both limits, roughly speaking in the region $\lambda = O(1)$. The critical behavior of the interacting theory is examined in the next section.

6.4 Critical Behavior in the Interacting Theory

For the following measurements, a tuning of β was necessary in order to achieve the desired values of mL. We have checked the achieved values of mL in the actual measurements for each simulation and in most cases hit the tuning target within one sigma, and in all cases within two sigma. The measured values of mL are presented in Appendix B.2.2.

In several cases, our results were better approximated by a fit of the type

$$\tau_{int} = \alpha + \beta \times \log(L). \tag{6.6}$$

than with the power law (6.1). To make clear which fit was used, we will in this section always list the complete fit function.

6.4.1 Two Dimensions

Results are represented in Figure 6.7 and the corresponding data can be found in Appendix B.2.2. We find much shorter autocorrelations and more diverse behavior compared to the Gaussian model in two dimensions. We find shorter autocorrelation times and lower critical exponents for all observables at $mL = 1$. The susceptibility here even shows critical speeding up.

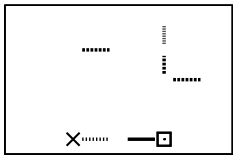
Neither an power law nor a logarithmic fit resulted in a good approximation to the behavior of $G(0)$. Since its values approach the completely uncorrelated case of $\tau_{int} = 0.5$ this could be due to a saturation in the vicinity of this value.

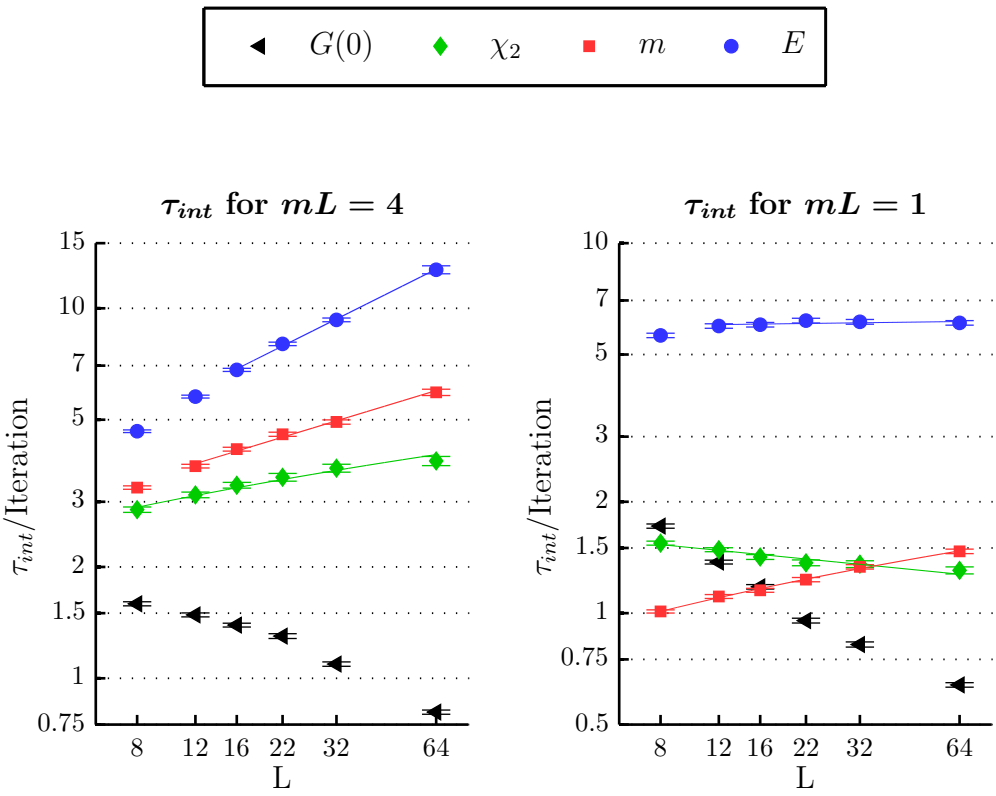
6.4.2 Three Dimensions

The graphical representation of the data, which is listed in Appendix B.2.2, is shown in Figure 6.8 with fit parameters. Measured autocorrelation times are comparable to those of the two dimensional case.

6.4.3 Four Dimensions

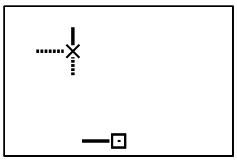
The results in the four dimensional theory are represented in Figure 6.9, and values can be found in Appendix B.2.2. Measured autocorrelation times are of similar length as in two and three dimensions.

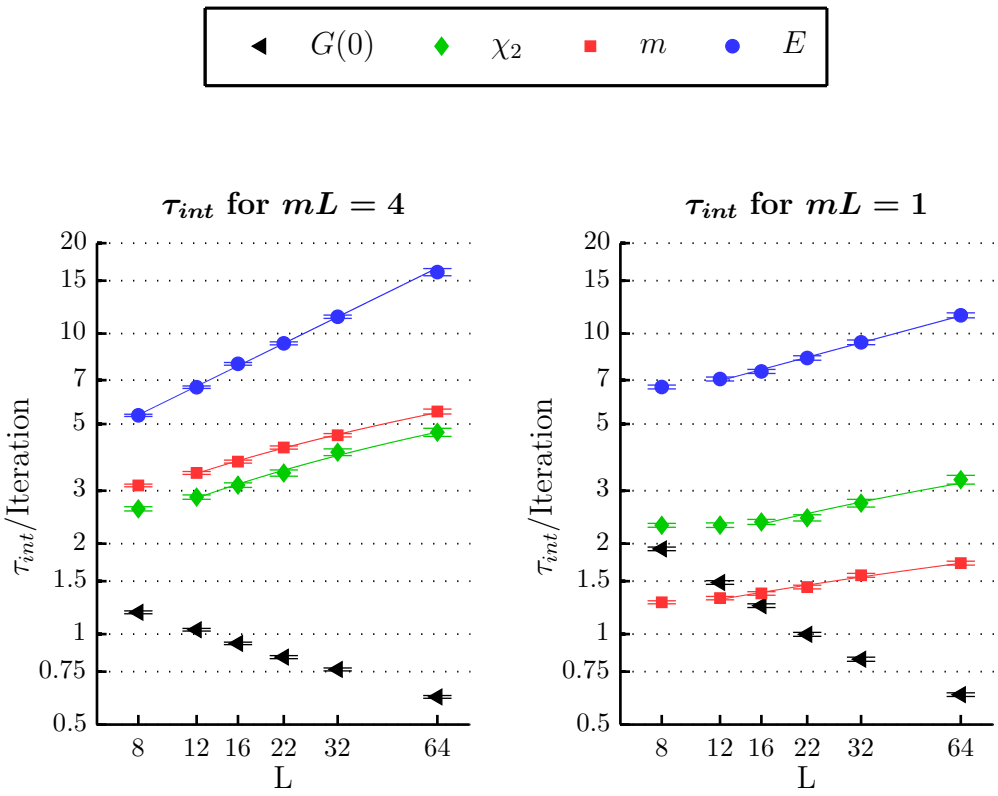




mL	Observable	Fit	χ^2/dof
4	χ_2	$1.7(1) + 0.54(4) \times \log(L)$	0.841114
4	m	$1.93(3) \times L\hat{}[0.27(1)]$	1.375721
4	E	$1.98(5) \times L\hat{}[0.44(1)]$	0.693817
1	χ_2	$1.84(3) \times L\hat{}[-0.08(1)]$	0.930718
1	m	$0.54(2) + 0.222(9) \times \log(L)$	0.375631
1	E	$5.8(2) + 0.06(6) \times \log(L)$	0.597387

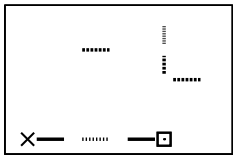
Figure 6.7: Critical dynamical behavior in the two-dimensional interacting theory at $\lambda = 0.5$ for four observables. Fit functions are listed in the table. Lines connect points that were used for fitting.

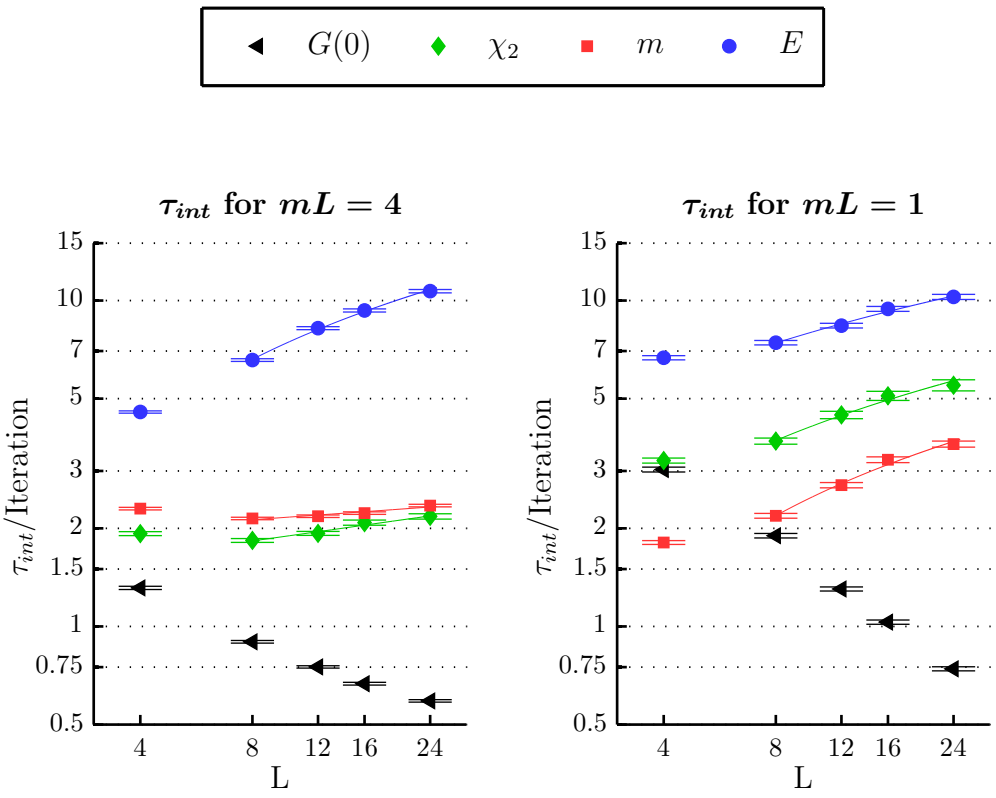




mL	Observable	α	z	χ^2/dof
4	χ_2	$0.0(2) + 1.11(7) \times \log(L)$	0.552360	
4	m	$0.4(1) + 1.21(5) \times \log(L)$	0.238090	
4	E	$1.73(2) \times L\hat{}[0.541(8)]$	0.896484	
1	χ_2	$0.6(2) + 0.61(7) \times \log(L)$	1.161015	
1	m	$0.68(5) + 0.24(1) \times \log(L)$	0.685986	
1	E	$3.33(4) \times L\hat{}[0.29(1)]$	0.445104	

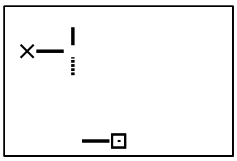
Figure 6.8: Critical dynamical behavior in the three-dimensional interacting theory at $\lambda = 0.5$ for four observables. Fit functions are listed in the table. Lines connect points that were used for fitting.





mL	Observable	Fit	χ^2/dof
4	χ_2	$1.30(5) \times L\hat{}[0.16(2)]$	0.585829
4	m	$1.79(3) \times L\hat{}[0.08(1)]$	1.450517
4	E	$-1.3(2) + 3.8(1) \times \log(L)$	0.795077
1	χ_2	$0.0(4) + 1.7(1) \times \log(L)$	0.511429
1	m	$-0.6(1) + 1.37(6) \times \log(L)$	1.277154
1	E	$1.8(4) + 2.6(1) \times \log(L)$	0.655217

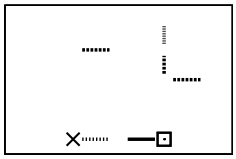
Figure 6.9: Critical dynamical behavior in the four-dimensional interacting theory at $\lambda = 0.5$ for four observables. Fit functions are listed in the table. Lines connect points that were used for fitting.

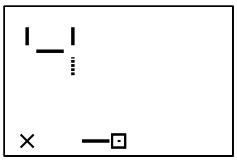


6.4.4 Discussion

For every dimension we have found much shorter autocorrelation times than in the Gaussian case. The highest autocorrelation is always exhibited by the energy, and we find strongly varying behavior among the observables, which we expected after our heuristic analysis of the Gaussian case. Furthermore, autocorrelations are of similar size, for $mL = 1$ and $mL = 4$. This stands in strong contrast to the Gaussian model, where we have found autocorrelation times differing by about one order of magnitude, depending on the systems correlation length. We have above expressed our expectation of a monotonous progression of autocorrelations along the λ-axis. Our brief analysis of the three dimensional case as shown in Figure 6.6 has further supported this assumption. If it holds, the critical dynamical behavior of the worm algorithm for $\lambda \geq 1/2$ is very favorable.

Yet it is important to note, that the critical behavior and consequently the performance depends on the observable of interest. For example, the two-point function $G(0)$ has in the interacting theory shown critical speeding up in every set of measurements. Autocorrelation times of susceptibility and mass grow slowly with the lattice size, at a logarithmic rate or with a critical exponent below 0.3. Finally, the energy shows the longest integrated autocorrelation times and highest critical exponents which however remain below 0.6.





7 Estimator for the Renormalized Coupling

In this chapter, we will present and test an estimator for the renormalized coupling, which uses two independent replica of a two-insertion ensemble. The main idea is to remove the line of active links connecting the insertions from the first ensemble and insert it into the second, thereby creating one vacuum configuration and one configuration that corresponds to a four-insertion ensemble. This is what we need to calculate four-point functions.

However, in our present form of the partition function, we sum over possible configurations of the link field. There are many possible paths connecting the insertions. In order to compare partition functions after removing or adding paths to an ensemble, we need to describe configurations in the "language" of graphs and loops. We will do this by generalizing the strong coupling expansion presented in Section 4.1 to the N-component theory, which yields the graph language in a natural way. We follow [32] where results from [33] were generalized to finite couplings.

7.1 Strong Coupling Expansion of N Component ϕ^4 Theory

We consider from now on an N-component scalar field $\phi_\alpha(x)$. Functions $Z_n(x_1, x_2, ..., x_n)$ in (4.2) are then replaced by functions of the form

$$Z_n(x_1, \alpha_1; x_2, \alpha_2; ...; x_n, \alpha_n) = \int D_\lambda\phi \; e^{\beta \sum_{l=\langle xy\rangle} \phi(x)\cdot\phi(y)} \phi_{\alpha_1}(x_1)\phi_{\alpha_2}(x_2)...\phi_{\alpha_n}(x_n) \quad (7.1)$$

where $\phi(x) \cdot \phi(y)$ denotes a scalar product and $\alpha_i \in \{1, ..., N\}$. The normalized integration measure is given by

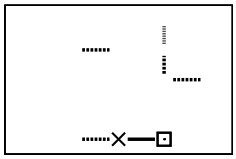
$$D_\lambda\phi = \prod_x \left\{ K(N,\lambda) d^N[\phi(x)] e^{-\phi(x)^2 - \lambda\left(\phi(x)^2-1\right)^2} \right\}, \quad K(N,\lambda) \leftrightarrow \int D_\lambda\phi = 1. \quad (7.2)$$

The right hand side of (7.1), including the measure, is $O(N)$-invariant, allowing us to split it up by writing

$$Z_n(x_1, \alpha_1; x_2, \alpha_2; ...; x_n, \alpha_n) = \sum_{a=1}^{A(n)} = T_n^a(\alpha_1, \alpha_2, ..., \alpha_n) X_n^a(x_1, x_2, ..., x_n) \quad (7.3)$$

with the grouping tensor T_a^n. Each of its elements contains one possible contraction of the $\alpha_1, \alpha_2, ..., \alpha_n$ into pairs. There are

$$A(n) = (n-1)!! \quad (7.4)$$



ways to do this and accordingly $a = 1, 2, ..., A(n)$. We expand around $\beta = 0$, this time using a generating functional $G_{N,\lambda}$ with sources j_α, which is defined by

$$\int D_\lambda \phi e^{\sum_x j(x)\phi(x)} = \prod_x G_{N,\lambda}(j(x)). \tag{7.5}$$

The generating functional has an expansion depending only on $(j \cdot j)$ due to $O(N)$ invariance

$$G_{N,\lambda}(j) = \sum_{l=0}^{\infty} C[l; N, \lambda] \frac{(j \cdot j)^l}{2^l l!}. \tag{7.6}$$

Forr $N = 1$ the expansion coefficients are related to the previously discussed weights c by

$$C[l; 1, \lambda] = \frac{2^l l!}{(2l)!}\, c(2l) = \frac{1}{(2l-1)!!}\, c(2l). \tag{7.7}$$

Using the generating functional, we express the expansion of (7.1) around $\beta = 0$ as

$$Z_n = \prod_{i=1}^{n} \left\{ \frac{\partial}{\partial j_{\alpha_i}(x_i)} \right\} \sum_{\{k\}} \prod_{l=\langle xy \rangle} \left\{ \frac{\beta^k(l)}{k(l)!} \left[\frac{\partial}{\partial j(x)} \frac{\partial}{\partial j(y)} \right]^{k(l)} \right\} \prod_z G_{N,\lambda}(j(z))|_{j \equiv 0} \tag{7.8}$$

where the the sum over $\{k\}$ runs over all possible configurations of the link field. For each such configuration k, the field $d(x)$ counts the derivatives acting on site x,

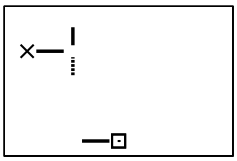
$$d(x) = \sum_{l, \partial l \ni x} k(l) + \sum_{m=1}^{n} \delta_{x_m, x}. \tag{7.9}$$

The $O(N)$ symmetry of the measure limits configurations to those providing even values $d(x)$ at all sites. From the terms that make up $G_{N,\lambda}$, only those with $(j \cdot j)^{d(z)/2}$ at each site will be nonzero after taking the derivatives and setting the sources to zero. Taking the derivatives results in sums of products of delta functions, representing each possible pairwise contraction of $O(N)$ indices at each site. The total number of terms is

$$\mathcal{M}_0[\{x_i\}; k] = \prod_z d(z)!. \tag{7.10}$$

Each term has a graphical representation similar to the one used until now for $N = 1$ (for example, in Figure 4.2), but now including the connection of lines at sites. Insertions must be pairwise connected by loops that are not closed geometrically, but in terms of the $O(N)$ contraction. There are $A(n)$ possible pairings, and each distinct pairing appears once in the grouping tensor T. All other lines must be arranged into geometrically closed loops, and each contributes a factor N to Z_n.

This factor is what makes it necessary to describe configurations as collections of loops for $N > 1$. For $N = 1$, its contribution is trivial. A configuration is completely described by the set $\{k; \{x_i\}\}$ which is what we have done until now. For $N > 1$, configurations in one pairing class a have different weights depending on the arrangement of lines into



loops. We call each such arrangement a graph Λ and the number of closed loops of this graph $|\Lambda|$.

Among the $\mathcal{M}_0$ terms in Z_n for each configuration of the link field, each graph Λ is represented

$$\mathcal{M}[\Lambda] = \frac{1}{\mathcal{S}[\Lambda]} \prod_l k(l) \prod_z \frac{d(z)!}{(d(z)-1)!!} \tag{7.11}$$

times. The product over sites is related to the possible ways to connect the lines surrounding site z with each other. The product over links compensates for terms resulting from permutations of lines on the link k. This leads in some cases to an overcounting which is corrected by the factor $\mathcal{S}[\Lambda]$.

X_n^a can now be organized as a sum over graphs. We denote the set of all possible graphs for a given insertion set $\{x_1, x_2, ..., x_n\}$ and pairing class a as $\mathcal{L}_n^a(\{x_i\})$. For arbitrary N, the final result is

$$X_n^a(x_1, x_2, ..., x_n) = \sum_{\Lambda \in \mathcal{L}_n^a(\{x_i\})} W(\Lambda) N^{|\Lambda|} \tag{7.12}$$

with

$$W(\Lambda) = \beta^{\sum_l k(l)} \frac{1}{\mathcal{S}[\Lambda]} \prod_z C[d(z)/2; N, \lambda]. \tag{7.13}$$

The fields $k(l)$ and $d(z)$ are now functions of the graph Λ. An algorithm sampling these graphs can be formulated as done in [33] for the limit $\lambda \to \infty$.

We will now return to $N = 1$ and use the combinatorics discussed above to reorder the sum over configurations $\{k\}$ appearing in our previous result (4.8). Combining (7.10) with (7.11), we formulate an identity for an arbitrary configuration $(k, \{x_i\})$,

$$1 = \prod_z \frac{1}{d(z)!} \underbrace{\sum_{\Lambda_{k,\{x_i\}}} \mathcal{M}[\Lambda]}_{\mathcal{M}_0} = \sum_{\Lambda_{k,\{x_i\}}} \frac{1}{\mathcal{S}[\Lambda]} \prod_l k(l)! \prod_z \frac{1}{(d(z)-1)!!} \tag{7.14}$$

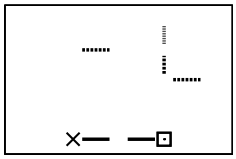
where $\Lambda_{k,\{x_i\}}$ are all possible graphs compatible with the given link values and insertions $(k, \{x_i\})$. We insert it into (4.8) and find

$$Z(x_1, x_2, ..., x_n) = \sum_{\{k\}} \prod_l \frac{\beta^{k(l)}}{k(l)!} \prod_z c(d(z)) \times 1 \tag{7.15}$$

$$= \sum_{\{k\}} \beta^{\sum_l k(l)} \prod_z \frac{c(d(z))}{(d(z)-1)!!} \sum_{\Lambda_{k,\{x_i\}}} \frac{1}{\mathcal{S}[\Lambda]}. \tag{7.16}$$

$$= \sum_{\{k\}} \sum_{\Lambda_{k,\{x_i\}}} W(\Lambda) = \sum_a \sum_{\Lambda \in \mathcal{L}_n^a(\{x_i\})} W(\Lambda) \tag{7.17}$$

which is the case $N = 1$ of (7.12). This is the result we were looking for. It allows us to compare configurations after adding or removing a given path in a simulation of the



one-component theory.

7.2 Four-Point Functions

We return to the one-component scalar field φ. According to (4.1), in order to compute four-point functions, we need $X(x_1, x_2, x_3, x_4)$ as well as $Z(\emptyset)$. We will consider for the moment the product of these two functions which, with (7.17), reads

$$X(x_1, x_2, x_3, x_4)Z(\emptyset) = \sum_{\bar{\Lambda} \in \mathcal{L}_4(x_1, x_2, x_3, x_4)} W(\bar{\Lambda}) \sum_{\bar{\Lambda}' \in \mathcal{L}_0} W(\bar{\Lambda}'). \tag{7.18}$$

This product can be related to $Z(x_1, x_2)Z(x_3, x_4)$ by removing the line between x_3 and x_4 from $\bar{\Lambda}$ and adding it to $\bar{\Lambda}'$. This defines a mapping

$$\mathcal{L}_4(x_1, x_2, x_3, x_4) \times \mathcal{L}_0 \ni (\bar{\Lambda}, \bar{\Lambda}') \leftrightarrow (\Lambda, \Lambda')\mathcal{L}_2(x_1, x_2) \times \mathcal{L}_2(x_3, x_4). \tag{7.19}$$

For each of the graphs $\Lambda, \Lambda', \bar{\Lambda}, \bar{\Lambda}'$, we denote corresponding link values $k(l)$, $k'(l)$, $\bar{k}(l)$, $\bar{k}'(l)$ and auxiliary fields $d(l), d'(l), \bar{d}(l), \bar{d}'(l)$. The mapping conserves

$$|\bar{\Lambda}| = |\Lambda|, |\bar{\Lambda}'| = |\Lambda'|, \;\; \bar{k}(l) + \bar{k}'(l) = k(l) + k'(l), \;\; \bar{d}(l) + \bar{d}'(l) = d(l) + d'(l) \tag{7.20}$$

as well as symmetry factors, since only lines connecting insertions are changed,

$$\mathcal{S}[\bar{\Lambda}] = \mathcal{S}[\Lambda], \;\; \mathcal{S}[\bar{\Lambda}'] = \mathcal{S}[\Lambda']. \tag{7.21}$$

Using the quantities defined by this mapping

$$X(x_1, x_2, x_3, x_4)Z(\emptyset) = \sum_{\Lambda \in \mathcal{L}_2(x_1, x_2)} \sum_{\Lambda' \in \mathcal{L}_2(x_3, x_4)} W(\Lambda)W(\Lambda')R(\Lambda, \Lambda') \tag{7.22}$$

with

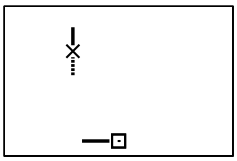
$$R(\Lambda, \Lambda') = \frac{W(\bar{\Lambda})W(\bar{\Lambda}')}{W(\Lambda)W(\Lambda')} \tag{7.23}$$

holds.

The line connecting x_3 and x_4 can be described by the passed sites z_i and the corresponding multiplicities ν_i indicating the number of self-intersections of the line at site z_i. Evaluating R for a given line (z_i, ν_i) and given fields d, d' yields

$$R(\Lambda, \Lambda') = \sum_{(z_i, \nu_i)} \frac{c(d(z_i) + 2\nu_i)}{c(d(z_i))} \frac{(d(z_i) - 1)!!}{(d(z_i) + 2\nu_i - 1)!!} \frac{c(d'(z_i) - 2\nu_i)}{c(d'(z_i))} \frac{(d'(z_i) - 1)!!}{(d'(z_i) - 2\nu_i - 1)!!}. \tag{7.24}$$

For $\lambda \to \infty$, we have found $c \equiv 1$ and only the double factorials survive. With our previous result (4.35), we evaluate R for the free case and find $R \equiv 1$.



Analogously to (4.17), we define an enlarged ensemble of two two-insertion replicas

$$\mathcal{Z}^2 = \sum_{u,v} Z(u,v) \sum_{u',v'} Z(u',v') = \sum_{u,v,\{k\},u',v',\{k'\}} \sum_{\Lambda_{k,u,v}} \sum_{\Lambda'_{k',u',v'}} W(\Lambda)W(\Lambda') \tag{7.25}$$

and the corresponding average

$$\langle\langle A \rangle\rangle_{2\times 2} = \frac{1}{\mathcal{Z}^2} \sum_{u,v,u',v'} \sum_{\{k\},\{k'\}} \sum_{\Lambda_{k,u,v}} \sum_{\Lambda'_{k',u',v'}} W(\Lambda)W(\Lambda')A(\Lambda,\Lambda'). \tag{7.26}$$

Summing over the insertions in (7.22) yields

$$\sum_{u,v,u',v'} Z(u,v,u',v')Z(\emptyset) = 3\,\mathcal{Z}^2 \left\langle\langle R(\Lambda,\Lambda') \rangle\right\rangle_{2\times 2}, \tag{7.27}$$

where the factor three stems from the summation over the $(4-1)!!$ possible pairwise connections of the four insertions. To estimate four-point functions we still need $Z(\emptyset)$, which can be averaged from the two ensembles as before,

$$\langle\langle \delta_{u,v}\tilde{c}(d(u)) \rangle\rangle^2 = \left\langle\langle \delta_{u,v}\tilde{c}(d(u))\delta_{u',v'}\tilde{c}(d'(u')) \rangle\right\rangle_{2\times 2} = V^2 \frac{Z(\emptyset)^2}{\mathcal{Z}^2} = \frac{1}{\chi_2^2}. \tag{7.28}$$

The last part follows from (4.50) and will be useful in the next subsection. Combining these formulas yields the estimator for four-point functions

$$\frac{Z(x_1,x_2,x_3,x_4)}{Z(\emptyset)} = 3V^2 \frac{\left\langle\langle \delta_{u,x_1}\delta_{v,x_2}\delta_{u',x_3}\delta_{v',x_4} R(\Lambda,\Lambda') \rangle\right\rangle_{2\times 2}}{\left\langle\langle \delta_{u,v}\tilde{c}(d(u))\delta_{u',v'}\tilde{c}(d'(u')) \rangle\right\rangle_{2\times 2}}. \tag{7.29}$$

The numerator can be improved by permuting u, v, u', v' and using translation invariance:

$$V Z(x_1,x_2,x_3,x_4) \tag{7.30}$$

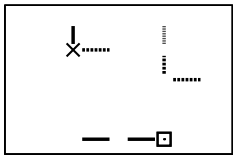
$$= \frac{1}{8} \left\langle\langle (\delta_{v,x_1-x_2}\delta_{u',x_1-x_3}\delta_{v',x_1-x_4} + \text{ 23 permutations of } u,v,u',v')\, R(\Lambda,\Lambda) \rangle\right\rangle_{2\times 2} \tag{7.31}$$

7.2.1 Susceptibilities and Renormalized Coupling

The renormalized coupling g_R was defined in (2.53). The appearing connected four-point susceptibility contains a sum over four-point functions, which we relate to $\langle\langle R \rangle\rangle_{2\times 2}$ using (7.27) and translation invariance

$$\sum_{x,y,z} \langle \varphi(0)\varphi(x)\varphi(y)\varphi(z) \rangle = \frac{1}{V} \sum_{x_1,x_2,x_3,x_4} \frac{Z(x_1,x_2,x_3,x_4)}{Z(\emptyset)} \tag{7.32}$$

$$= \frac{3}{V} \left(\frac{\mathcal{Z}}{Z(\emptyset)} \right)^2 \langle\langle R(\Lambda,\Lambda') \rangle\rangle_{2\times 2}. \tag{7.33}$$



With (7.28), the dependence on $\mathcal{Z}$ can be eliminated from this equation, leaving us with

$$\sum_{x,y,z} \langle \varphi(0)\varphi(x)\varphi(y)\varphi(z)\rangle = 3V\chi_2^2 \left\langle\langle R(\Lambda,\Lambda')\rangle\right\rangle_{2\times 2}. \tag{7.34}$$

This result leads to a simple estimator for the renormalized coupling, which reads

$$g_R = -\frac{\chi_{4,c}}{\chi_2^2} m^D = 3\left[1 - \left\langle\langle R(\Lambda,\Lambda')\rangle\right\rangle_{2\times 2}\right](mL)^D. \tag{7.35}$$

7.2.2 Improved Estimators

In our simulations, we sample configurations $(\{k\}, u, v)$ instead of graphs. In order to be able to estimate observables, we need to integrate the sums over graphs in (7.26) as well as the symmetry factors appearing in $W[\Lambda]$ into the observable A. This will result in a prescription of how to choose the line connecting insertions for given $(u, v; k)$ and $(u', v'; k')$. We reconsider the definition of averages, reorder the terms and insert factorials of link values in numerator and denominator,

$$\langle\langle A\rangle\rangle_{2\times 2} = \frac{1}{\mathcal{Z}^2} \sum_{u,v,\{k\}} \sum_{u',v',\{k'\}} \prod_l w(k(l))w(k'(l)) \prod_z c(d(z))c(d'(z)) \tag{7.36}$$

$$\times \underbrace{\sum_{\Lambda_{k,u,v}} \sum_{\Lambda'_{k',u',v'}} \frac{k(l)!}{(d(z)-1)!!} \frac{1}{\mathcal{S}[\Lambda]} \frac{k'(l)!}{(d'(z)-1)!!} \frac{1}{\mathcal{S}[\Lambda']} A(\Lambda,\Lambda')}_{\mathcal{A}(u,v,k,u',v',k')}. \tag{7.37}$$

The factors appearing in the improved estimator are multiplicities of graphs,

$$\mathcal{A}(u,v,k,u',v',k') = \sum_{\Lambda_{u,v,k}} \underbrace{\frac{\mathcal{M}[\Lambda]}{\mathcal{M}_0[u,v;k]}}_{\mathcal{P}[\Lambda]} \sum_{\Lambda'_{u',v',k'}} \underbrace{\frac{\mathcal{M}[\Lambda']}{\mathcal{M}_0[u',v';k']}}_{\mathcal{P}[\Lambda']} A(\Lambda,\Lambda'). \tag{7.38}$$

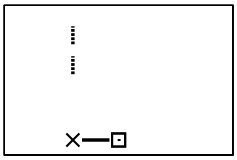
The values $\mathcal{P}[\Lambda]$ are the normalized probabilities to construct the active line connecting insertions in the graph Λ by randomly pairing lines at sites.

The observable R is defined in terms of the fields d, d' and the set (z_i, ν_i) given by the graph Λ'. It has equal values for all $\Lambda_{u,v,k}$. Hence, for R the first sum in (7.38) can be executed, and the improved estimator for R then reads

$$\mathcal{R}(u,v,k,u',v',k') = \sum_{\Lambda'_{u',v',k'}} \mathcal{P}(\Lambda') R(u,v,k,\Lambda'). \tag{7.39}$$

7.3 Testing the Estimator

Before we present our numerical results, we build up some expectations on the estimator $\langle\langle R\rangle\rangle_{2\times 2}$. We have already noted that $R \equiv 1$ in the free case. With (7.34), it follows



immediately that

$$\chi_{4,c} \equiv 0 \qquad \text{if } \lambda = 0, \tag{7.40}$$

and the renormalized coupling vanishes. The estimator yields the exact result without errors of fluctuations. On the other hand, the number of factors in (7.24) grows with the length of the displaced line. This may result in large fluctuations as soon as factors unequal one appear. We expect the number of contributing graphs to grow with the criticality of the system, consequently fluctuations of R might increase towards the critical point.

At fixed mL, we expect the renormalized coupling to grow as $\lambda \to \infty$. Hence, the mean of R should shrink in this limit. R is per definition a product of positive numbers and therefore bounded from below by zero. On the other hand, at least in the Ising limit, R is not bounded from above.

To show this, we consider a configuration (u, v, k, u', v', k') with $u' = v'$. Then, there is a graph Λ' where the insertions are directly connected with each other, and the (in this case rather short) line which we displace from Λ' to Λ is described by $(z_i, \nu_i) = (u', 1)$. Inserting this into (7.24) for $\lambda = \infty$ yields

$$R(\Lambda, \Lambda') = \frac{d'(u) - 1}{d(u) + 1}. \tag{7.41}$$

The Lebowitz inequality states that $\langle\langle R \rangle\rangle_{2\times 2} \leq 1$. We can read off from equation (7.41) that this does not hold for individual estimates of R. In the above example, for $d'(u) < d(u) + 2$ we have $R > 1$.

Summarizing our expectations for the Ising case, it seems not improbable to frequently find estimates of R greater than one, which can only be compensated by numerous estimates very close to zero, in order to provide a small mean of R. With this in mind, we will need to gain an impression of the occurring distributions of R before we apply our usual error analysis.

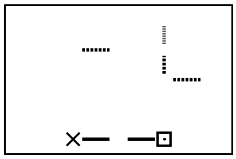
7.3.1 Implementation

To estimate $\langle\langle R \rangle\rangle_{2\times 2}$, we have simulated two replica of the worm algorithm in parallel. After one iteration in both replica, a line connecting the insertions in one ensemble was constructed. To keep track of lines visited during the selection of the line, a second link field $v(l)$ was introduced. At the beginning of construction, it was set to zero for all links.

Construction starts at u. Each step consists of selecting one of the $d_0(x)$ lines connected to the current site x with probability

$$p_{\hat{l}} = \frac{k(\hat{l}) - v(l)}{\sum_{l \ni \partial x} (k(l) - v(l)) + \delta_{x,v}}, \tag{7.42}$$

moving across this link and incrementing its value $v(l)$ by one. Each visited site, including the starting point u, must be included into the z_i and its multiplicity ν_i must be



incremented at each visit.

If v coincides with the current site x, the probability to complete the construction is given by

$$p_v = \frac{\delta_{x,v}}{\sum_{l \ni \partial z} (k(l) - v(l)) + \delta_{x,v}}. \tag{7.43}$$

After the path is selected, R is calculated according to (7.24).

7.3.2 Distribution of Estimates

We begin to close the free case where we expect the least problematic distributions. For these tests, we simulate on lattices of 16^3 sites. In Chapter 5, we have determined the values of β to achieve $mL = 1$ and $mL = 4$. We measure $5 \cdot 10^5$ iterations of two replica resulting in $5 \cdot 10^5$ estimates of R, and show the acquired histograms in Figure 7.1. Although we are very close to the free case, we already see a wide distribution at $mL = 4$.

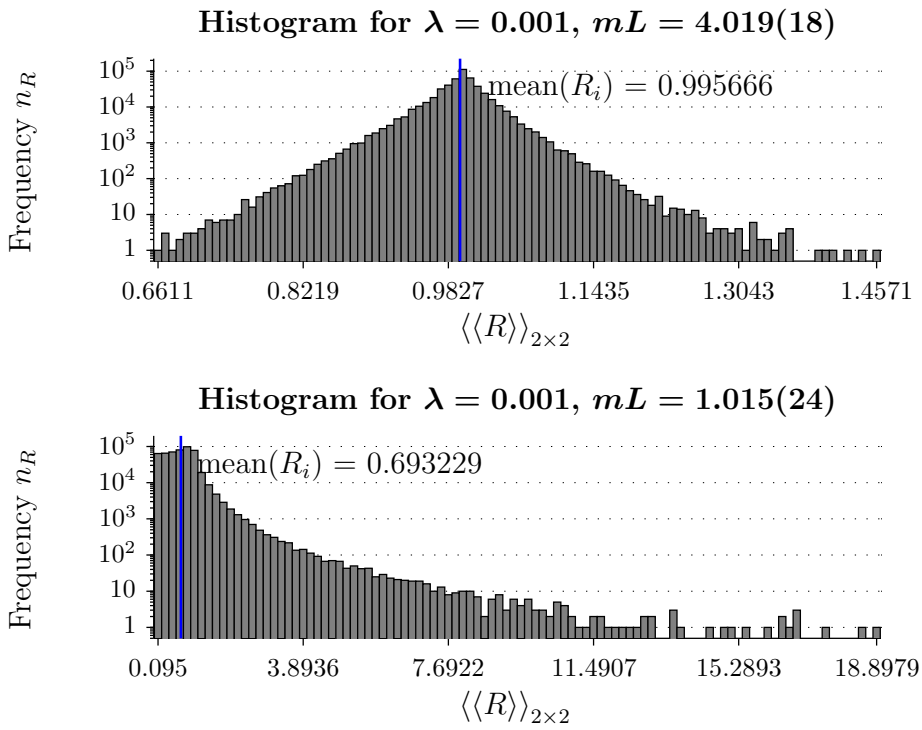
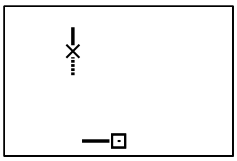


Figure 7.1: Histograms of $5 \cdot 10^5$ estimates of $\langle\langle R \rangle\rangle$ on a 16^3 lattice at $\lambda = 0.001$ and $mL = 1$ as well as $mL = 4$. Blue lines mark the overall mean.

The histogram appears almost symmetric, except for the highest occurring R, which form a tail. However, there are no outliers and we assume to have sufficient statistics on the tail for an error analysis. We compare results from an identical simulation with



Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
m^2	0.0642(17)	0.563(61)	0.06250(44)	0.611(65)
$z = mL$	4.055(56)	0.563(61)	4.000(14)	0.611(65)
ϕ^2	0.70423(23)	0.498(31)	0.70427(24)	0.539(58)
χ_2	47.410(0.8)	0.520(46)	48.229(0.3)	0.614(66)
conn χ_4 [/1e6]	-0.57(37)	0.502(44)	-0.1212(51)	0.496(31)

Table 7.1: Comparison of results for the connected four-point susceptibility and other observables. Simulation parameters are $\lambda = 0.001$, $\beta = 0.330620$, $D = 3$, $L = 16$.

those of a Metropolis simulation with the same parameters in Table 7.1. The results are compatible. We clearly see the large errors in the Metropolis estimate of $\chi_{4,c}$ caused by connected cancelation, and a much higher precision achieved with the worm algorithm.

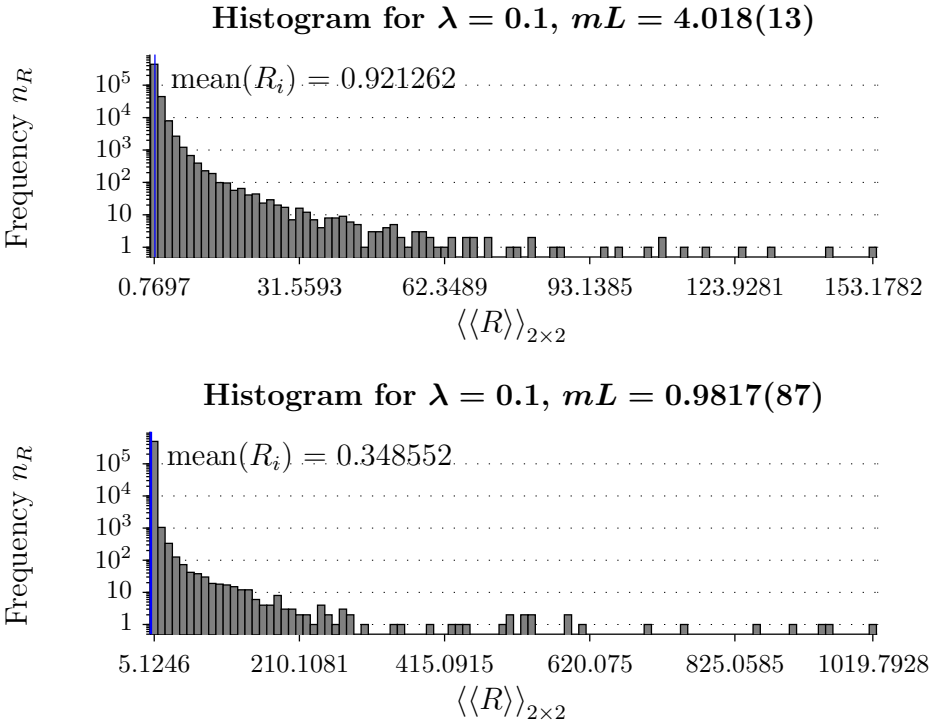
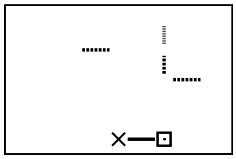


Figure 7.2: Histograms of $5 \cdot 10^5$ estimates of $\langle\langle R \rangle\rangle$ on a 16^3 lattice at $\lambda = 0.1$ and $mL = 1$ as well as $mL = 4$. Blue lines mark the overall mean.



For $mL = 1$ however, the histogram has shifted. It shows a pronounced tail which falls off only slowly. The highest outlier seen in our measurement is two orders of magnitude larger than the mean. For a solid estimation of the decay of the tail, more measurements would be necessary, however, the qualitatively different distribution suggests a much higher variance than in the case $mL = 4$. This confirms our apprehension that large fluctuations could make this estimator less efficient for critical systems.

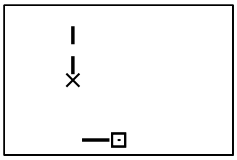
We repeat this investigation at $\lambda = 0.1$ in order to check if we can advance further towards stronger couplings at least for $mL = 4$. The histograms are shown in Figure 7.2.

The distributions of estimates are similar to the one at $mL = 1, \lambda = 0.001$. In the simulation with $mL = 1$, the highest outlier is four orders of magnitude larger than the mean. Removing this single measurement from the data would effect a change of the mean of roughly three percent. To be able to apply our usual statistical tools, we would demand the relative frequency of such contributions to be much greater than the total number of estimates. Even very close to the free theory at $\lambda = 0.1$, we are unable to satisfy this condition. For $mL = 4$, the range of obtained estimates is one order of magnitude smaller. However, we again have very high estimates wich occurred only once among our estimates of R. The tendency towards stronger couplings is clear. The tail grows towards higher estimates of R, resulting in higher variances and longer measurements needed to control them.

This result is not implausible. The factors in (7.24) can become smaller or larger than one if $d(x)$ and $d'(x)$ have very different values. A line visiting ten sites and resulting in a factor two at each site already results in an estimate of $O(10^3)$. Since R is only bounded from below, means of $O(1)$ can only result from an asymmetric distribution.

To visualize this, we have simulated on an 8^2 lattice at $\lambda = 1, \beta = 0.6$. We have independently measured $mL = 2.817(9)$ for these parameters. During one simulation containing 10^6 estimates of R, the configurations $(u, v; k)$ and $(u', v'; k')$ shown in Figure 7.3 occured. While the configuration $(u', v'; k')$ shows a strongly populated link field, there are only a few closed paths in the vacuum configuration (u, v, k). A line connecting u' and v' was constructed as described in Section 7.3.1. At the bottom of the figure, we show the resulting field $v(l)$ as a representation of that line. The mean of all 10^6 estimates of R was ≈ 0.64. The estimate resulting from the configurations and displaced line shown in the figure was approximately $2.4 \cdot 10^4$.

In summary, we have found, that with the computing power available to us, the variance of the estimator $\langle\langle R \rangle\rangle_{2\times 2}$ can be controlled only in the parameter space where $\lambda \ll 1$, $mL \gg 1$. This means that it can neither be used close to the continuum limit, nor can it be utilized in the non-perturbative region.



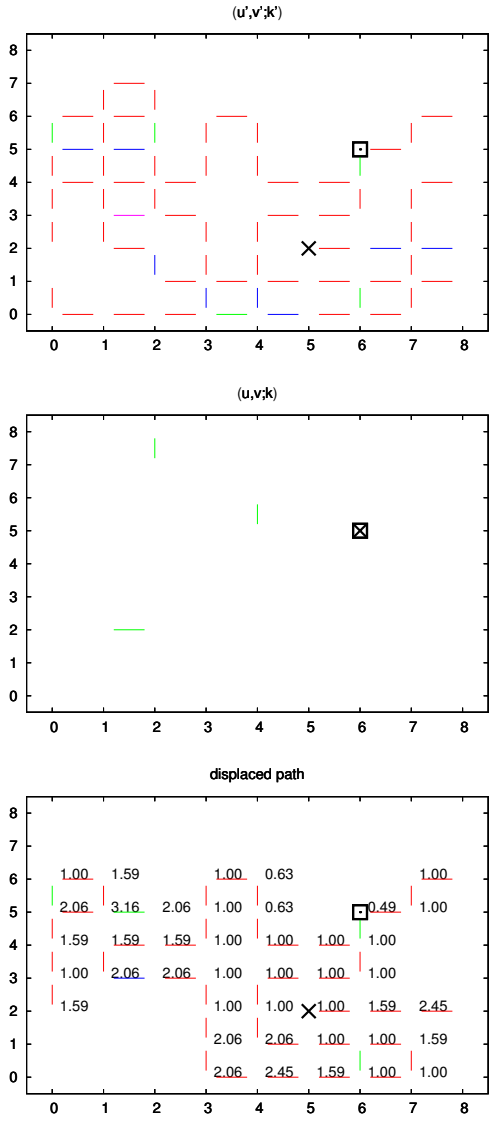
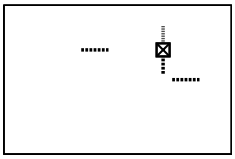
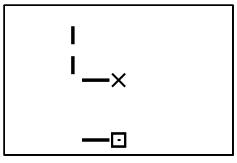


Figure 7.3: Configurations resulting in a large estimate of R. Configurations $(u, v; k)$ and $(u', v'; k')$ are shown in the top two plots. At the bottom, the line of active links $v(l)$ is represented, together with the factors of R, printed at the upper right hand of the corresponding site. The product of all contributions is $R \approx 2.4 \cdot 10^4$. Simulation parameters were $V = 8^2, \lambda = 1, \beta = 0.6$. Colors correspond to link values: $(\text{red}, \text{green}, \text{blue}, \text{magenta}) \mathrel{\widehat{=}} (1, 2, 3, 4)$. Box and cross represent u, v and u', v' respectively.





8 Conclusions and Outlook

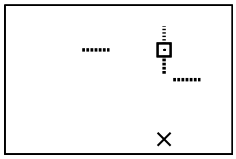
In this work, we have applied a recent approach to the simulation of critical systems to φ^4 theory. In the first chapter, we have motivated our interest in the new method as well as in the chosen field theory. In the following introductory chapters, φ^4 theory was discussed and the method of Monte Carlo simulation was introduced. There, we have also taken the opportunity to underline, that the dynamical critical behavior of an algorithm can severely limit the ability to approach the continuum limit of a theory.

In chapter four, the worm algorithm for φ^4 theory was presented. A Metropolis and a heat bath variant of the algorithm were described, and a method for the computation of weights appearing in the acceptance probabilities was presented.

After these preparations, we have commenced with numerical experiments. In the fifth chapter, our implementation of the algorithm was tested by comparing it to a standard Metropolis simulation. For the considered three-dimensional lattice, we observed that for a fixed ratio of correlation length and lattice extent, estimates gained precision towards the Ising limit. Also, worm estimates for the mass were always more precise than the Metropolis estimates, while in a region of small coupling strengths, two-point functions were calculated more precisely by means of the Metropolis algorithm. This confirmed results from previous studies, which found the performance of the worm algorithm to depend on the observable of interest.

In chapter six, we have presented our analysis of the dynamical behavior. In the two-dimensional Gaussian model, we observed strong critical slowing down for energy, second moment mass and two-point susceptibility. Critical exponents are greater than one and less than 1.65. On a lattice of linear extent 64 and equal correlation length, we recorded integrated autocorrelation times of 10^4 iterations length. This is, to our knowledge, the first observation of pronounced critical slowing down of the worm algorithm for observables like second moment mass and two-point susceptibility. It disappoints our hopes that the absence of critical slowing down observed in the Ising limit could survive for all coupling strengths. Critical exponents of the three mentioned observables are smaller for the three- and smaller again for the four-dimensional theory. In the latter case, the achieved precision depends only very weakly on the lattice size. We argued that this dynamical behavior is linked to the high energies typical for the low-dimensional Gaussian model and to strong correlations between observables in the free theory. We have predicted critical slowing down to be considerably reduced already for the weakly interacting theory.

We have also measured integrated autocorrelation times of two-point functions. In three dimensions, critical exponents are below 0.01, while in four dimensions we observe critical speeding up for these observables. This is due to cancellation of autocorrelations in the primary estimators.



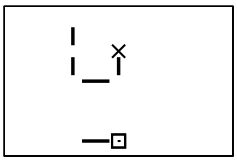
In the interacting theory at a coupling strength of $\lambda = 0.5$, measured autocorrelation times are on the considered lattices of order one or slightly longer, independent of the dimension. In a multitude of cases, we found that the data is best described with a logarithmic fit. Where this was not the case, the energy has shown the highest critical exponents, reaching 0.55 in three dimensions. Mass and susceptibility have critical exponents smaller then 0.3.

These results are similar to those observed in the Ising limit and very different from those we have measured in the Gauss model. We conclude, that the practical absence of critical slowing down observed in previous work in the Ising limit survives at least until $\lambda = 0.5$. In two and three dimensions, critical slowing sets in for smaller values of λ.

The algorithm we have examined so far is designed towards (and at first glance limited to) the extraction of two-point functions. In the seventh chapter we have presented and examined an estimator, which was designed to enable the extraction of four-point functions from two independent simulations using the original algorithm. Particularly, we hoped to gain precision by achieving analytically the subtraction of disconnected parts, as was acomplished in previous work for the Ising case [10]. Estimates obtained using this estimator however follow a distribution which is only under control close to the free theory and far away from the critical line, rendering the estimator inapplicable to cases of physical interest.

The limitation to two-point functions therefore remains. An interesting topic for future research could be the dynamical behavior of an algorithm sampling a four-insertion ensemble. The construction of such an ensemble and the extraction of observables are outlined in Appendix A.1. Although this approach would likely suffer from cancelation problems, it might be interesting to see if four-point functions extracted in this way show dynamical behavior similar to the one observed for two-point functions in this work.

Despite this limitation, we have shown that the worm algorithm is effective for the calculation of masses, two-point functions and two-point susceptibilities in φ^4 theory beyond infinite coupling. This makes it suitable, for example, for analyses of the phase transition in φ^4 theory. In this context, experiments in the broken phase would be of interest. The method has already been successfully generalized to the $O(N)$ sigma models, and a further generalization to the N component φ^4 theory seems straightforward. This might prove useful in future MC studies similar to [34].



A Additional Calculations and Results

A.1 Enlarging Further - Four Point Functions

The straightforward way to calculate four-point functions from a strong coupling simulation would be to simulate an ensemble with four field insertions. In Chapter 4, it was shown how an enlarged ensemble that sums over all possible sites of two field insertions samples $Z(x_1, x_2)$ and $Z(\emptyset)$ simultaneously. Analogously, one could define an ensemble that sums over all possible sites of four field insertions

$$\mathcal{Z}_4 = \sum_{s,t,u,v} Z(s,t,u,v) \tag{A.1}$$

and the corresponding average

$$\langle\langle A(k;s,t,u,v)\rangle\rangle_4 = \frac{1}{\mathcal{Z}_4} \sum_{s,t,u,v} \sum_{\{k\}} w(k) \prod_x c(d(x)) A(k;s,t,u,v). \tag{A.2}$$

Again, the constraint posed by $c(d(x))$ would hold and require $d(x)$ to be even at all sites. A configuration $Z(x_1, x_2, x_3, x_4)$ would in general consist of a collection of closed paths and two open paths ending at individual insertions.
If exactly two of the field insertions coincide, the two paths would be connected, and the configuration could be thought of as a collection of closed paths and *one* open path ending at the non-coinciding field insertions, i.e. a configuration also occuring in $Z(x_1, x_2)$.
Analogously to (4.15), partition functions can be compared:

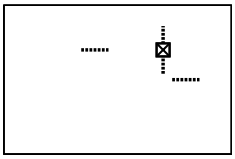
$$Z(x_1,x_2,x_3,x_3) = \sum_{\{k\}} w(k) \left(\prod_{x \neq x_3} c(d_0(x) + \delta_{x,x_1} + \delta_{x,x_2}) \right) c(d_0(x_3)+2) \tag{A.3}$$

$$Z(x_1,x_2) = \sum_{\{k\}} w(k) \left(\prod_{x \neq x_3} c(d_0(x) + \delta_{x,x_1} + \delta_{x,x_2}) \right) \tag{A.4}$$

$$\times\, c(d_0(x_3)+2)\tilde{c}(d_0(x_3)+2) \tag{A.5}$$

Or equivalently using (A.2),

$$\langle\langle \delta_{s,t}\delta_{x_1,u}\delta_{x_2,v}\tilde{c}(d(s))\rangle\rangle_4 = \frac{V}{\mathcal{Z}_4} Z(x_1,x_2). \tag{A.6}$$



If all four field insertions coincide the constraint forces a closed path configuration which relates to the proper partition function:

$$Z(x_1,x_1,x_1,x_1) = \sum_{\{k\}} w(k) \left(\prod_{x \neq x_1} c(d_0(x)) \right) c(d_0(x_1)+4) \tag{A.7}$$

$$Z(\emptyset) = \sum_{\{k\}} w(k) \left(\prod_{x \neq x_1} c(d_0(x)) \right) \tag{A.8}$$

$$\times\, c(d_0(x_1)+4)\tilde{c}(d_0(x_1)+4)\tilde{c}(d_0(x_1)+2). \tag{A.9}$$

Consequently, we can write

$$\langle\langle \delta_{s,t}\delta_{t,u}\delta_{u,v}\tilde{c}(d(s))\tilde{c}(d(s)-2)\rangle\rangle_4 = \frac{V}{\mathcal{Z}_4} Z(\emptyset). \tag{A.10}$$

A closed path configuration also arises if the field insertions coincide pairwise:

$$Z(x_1,x_1,x_2,x_2) = \sum_{\{k\}} w(k) \left(\prod_{x \neq x_1,x_2} c(d_0(x)) \right) c(d_0(x_1)+2)c(d_0(x_2)+2) \tag{A.11}$$

$$Z(\emptyset) = \sum_{\{k\}} w(k) \left(\prod_{x \neq x_1,x_2} c(d_0(x)) \right) \tag{A.12}$$

$$\times\, c(d_0(x_1)+2)\tilde{c}(d_0(x_1)+2)c(d_0(x_2)+2)\tilde{c}(d_0(x_2+2)). \tag{A.13}$$

Leading us to another estimator for the proper partition function:

$$\langle\langle \delta_{s,t}\delta_{u,v}(1-\delta_{t,u})\tilde{c}(d(s))\tilde{c}(d(u))\rangle\rangle_4 = \frac{V(V-1)}{\mathcal{Z}_4} Z(\emptyset) \tag{A.14}$$

Now we are ready to prescribe the extraction of four- and two-point functions from $\mathcal{Z}_4$ according to (4.1) where we point out the obvious improvement that arises from permutations of insertions. The definitions

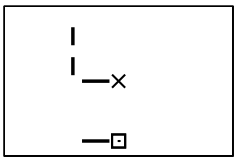
$$A_{0,1} = \delta_{s,t}\delta_{t,u}\delta_{u,v}\tilde{c}(d(u))\tilde{c}(d(u)-2)) \tag{A.15}$$

$$A_{0,2} = \frac{1}{3}\left(\delta_{s,t}\tilde{c}(d(s))\delta_{u,v}\tilde{c}(d(u))(1-\delta_{t,u}) + 2 \text{ more pairings of } s,t,u,v\right) \tag{A.16}$$

allow us to write the contribution of $Z(\emptyset)$ more compactly. With this, we finally find:

$$\langle\varphi(x_1)\varphi(x_2)\varphi(x_3)\varphi(x_4)\rangle = \frac{1}{24}\frac{\langle\langle \delta_{x_1,s}\delta_{x_2,t}\delta_{x_3,u}\delta_{x_4,v} + 23 \text{ permutations of } s,t,u,v\rangle\rangle_4}{\langle\langle A_{0,1}/V + A_{0,2}/V^2\rangle\rangle_4} \tag{A.17}$$

$$\langle\varphi(x_1)\varphi(x_2)\rangle = \frac{1}{6}\frac{\langle\langle \delta_{s,t}\delta_{x_1,u}\delta_{x_2,v}\tilde{c}(d(s)) + 5 \text{ permutations of } s,t,u,v\rangle\rangle_4}{\langle\langle A_{0,1} + A_{0,2}/V\rangle\rangle_4} \tag{A.18}$$



The calculation of sums over two- and four-point functions can again be reduced to the summation of distance histograms. In the proposed ensemble, all the observables discussed in this chapter could also be accumulated with analogously modified estimators. A Metropolis update algorithm could be very similar to the one described in section 4.4.1, where type II updates could be proposed whenever an even number of insertions coincide. Conventional wisdom suggests that the calculation of χ_4 in this way might be inefficient, due to cancellations of large similar numbers. On the other hand, we have found that the two-point functions in the two-insertions ensemble, which are constructed quite similarly, show very favorable dynamic behavior (see Chapter 6). This might justify some further study. However, in the line of this thesis we have considered in Chapter 7 a different approach, which was ultimately inspired by the work done by Aizenman in [17]. This approach held the potential to allow for a very efficient numerical estimation of g_R without further enlargement of the simulated ensemble, which justified in the opinion of the author, this decision.

A.2 Type III Step

We experiment with a third update step, which is executed independent of the location of the insertions. We hope that this step will help keeping the link field equilibrated and counteract the observed long correlations. We propose to add or subtract a value of two to a randomly chosen link $\hat{l}$. We call the sites connected by the link y_1 and y_2. The Metropolis acceptance probabilites are:

$$p_+ = \frac{\beta^2}{k(\hat{l})+1)(k(\hat{l})+2)} \frac{1}{\tilde{c}(d(y_1)+2)} \frac{1}{\tilde{c}(d(y_2)+2)} \tag{A.19}$$

$$p_- = \frac{k(\hat{l}))(k(\hat{l})-1)}{\beta^2} \tilde{c}(d(y_1))\tilde{c}(d(y_2)) \tag{A.20}$$

We choose at random if an incrementation or decrementation of the link is proposed. The Metropolis probabilities satisfy detailed balance, and ergodicity is already provided by the type I and II steps.

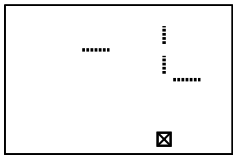
We add this step to our implementation. An update now consists of the successive execution of all three steps. We test this on a 16^2 lattice at $mL = 1$, were we know that $\mathcal{N}$ shows long autocorrelations. Without the third step we measure in a run of 10^6 Iterations

$$\tau_{int,\mathcal{N}} = 23(6) \cdot 10^3 \times 2 \text{ steps.} \tag{A.21}$$

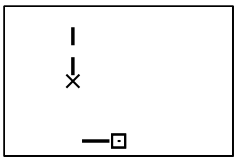
Close to the critical point the type II step is only very rarely executed. Adding a third step will roughly double the computational effort. With the extended implementation we measure 10^6 Iterations and find

$$\tau_{int,\mathcal{N}} = 18(6) \cdot 10^3 \times 3 \text{ steps.} \tag{A.22}$$

With doubled effort we have not succeeded in achieving a bisection of the autocorrelation



times. The new step does not decorrelate $\mathcal{N}$ faster then the original type II step.



B Numerical Results

B.1 Comparison with Standard Metropolis Simulation

B.1.1 $mL = 4$

$\lambda = 0.001, \beta = 0.330620, D = 3, L = 16$

Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	3.942(59)	0.580(62)	4.016(12)	0.478(43)
$G(0)$	0.70624(28)	0.576(62)	0.70589(24)	0.493(44)
χ_2	49.1(1.0)	0.578(62)	48.153(0.2)	0.475(42)

$\lambda = 0.010, \beta = 0.336446, D = 3, L = 16$

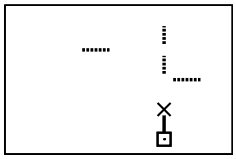
Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	3.948(56)	0.543(59)	4.015(12)	0.536(41)
$G(0)$	0.69222(21)	0.479(43)	0.69188(21)	0.593(58)
χ_2	48.27(93)	0.526(57)	47.14(27)	0.539(41)

$\lambda = 0.100, \beta = 0.366950, D = 3, L = 16$

Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	3.934(51)	0.619(77)	3.9973(93)	0.496(22)
$G(0)$	0.63037(12)	0.554(60)	0.63014(12)	0.489(37)
χ_2	44.00(76)	0.629(78)	43.15(19)	0.511(39)

$\lambda = 0.200, \beta = 0.380856, D = 3, L = 16$

Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	4.104(48)	0.459(29)	4.0091(89)	0.524(40)
$G(0)$	0.603770(88)	0.477(30)	0.60364(10)	0.483(30)
χ_2	40.04(59)	0.451(29)	40.99(16)	0.509(39)



$\lambda = 0.500, \beta = 0.388302, D = 3, L = 16$

Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	4.036(49)	0.486(30)	3.9953(76)	0.493(22)
$G(0)$	0.585605(65)	0.465(29)	0.585569(77)	0.506(31)
χ_2	39.40(61)	0.470(30)	39.96(14)	0.502(38)

$\lambda = 1.000, \beta = 0.370032, D = 3, L = 16$

Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	3.949(48)	0.501(31)	4.0104(70)	0.483(21)
$G(0)$	0.606696(51)	0.494(44)	0.606664(55)	0.484(30)
χ_2	42.00(66)	0.490(31)	41.05(12)	0.484(21)

$\lambda = 2.000, \beta = 0.322804, D = 3, L = 16$

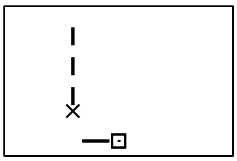
Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	3.992(54)	0.505(45)	3.9906(64)	0.491(31)
$G(0)$	0.683647(35)	0.485(43)	0.683603(37)	0.498(22)
χ_2	46.71(80)	0.495(54)	46.59(12)	0.463(29)

$\lambda = 5.000, \beta = 0.253849, D = 3, L = 16$

Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	4.182(85)	0.596(74)	4.0049(63)	0.546(48)
$G(0)$	0.851674(22)	0.535(67)	0.851687(14)	0.515(39)
χ_2	55.(14)	0.612(76)	57.95(16)	0.550(48)

$\lambda = 10.000, \beta = 0.230730, D = 3, L = 16$

Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	4.09(11)	0.80(12)	3.9941(61)	0.506(39)
$G(0)$	0.931214(14)	0.523(57)	0.9312404(65)	0.573(50)
χ_2	61.(21)	0.75(11)	63.44(16)	0.504(39)



B.1.2 $mL = 1$

$\lambda = 0.001, \beta = 0.334112, D = 3, L = 16$

Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	1.029(29)	1.51(30)	0.993(28)	1.88(29)
$G(0)$	0.8845(88)	1.46(27)	0.898(10)	1.70(26)
χ_2	711.(36.)	1.47(28)	766.(41.)	1.80(28)

$\lambda = 0.010, \beta = 0.341128, D = 3, L = 16$

Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
$z = mL$	0.993(15)	0.95(14)	1.007(13)	0.709(98)
$G(0)$	0.8538(33)	0.88(12)	0.8518(34)	0.82(11)
χ_2	685.(15.)	0.89(12)	671.(15.)	0.82(11)

$\lambda = 0.100, \beta = 0.375081, D = 3, L = 16$

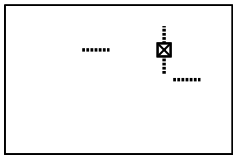
Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	0.9794(97)	0.636(79)	0.9982(86)	0.666(65)
$G(0)$	0.72245(80)	0.628(78)	0.72144(94)	0.713(76)
χ_2	515.(56)	0.636(79)	505.(56)	0.750(80)

$\lambda = 0.200, \beta = 0.390558, D = 3, L = 16$

Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	0.993(10)	0.633(88)	0.9868(73)	0.536(47)
$G(0)$	0.67498(57)	0.613(85)	0.67572(53)	0.524(46)
χ_2	458.(48)	0.624(87)	463.(38)	0.599(52)

$\lambda = 0.500, \beta = 0.400173, D = 3, L = 16$

Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	1.010(10)	0.555(69)	0.9989(64)	0.497(22)
$G(0)$	0.63479(36)	0.582(72)	0.63490(36)	0.565(50)
χ_2	410.(41)	0.571(71)	413.(28)	0.529(40)



$\lambda = 1.000, \beta = 0.383043, D = 3, L = 16$

Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	0.994(11)	0.80(12)	1.0067(66)	0.540(41)
$G(0)$	0.64281(27)	0.78(11)	0.64289(23)	0.524(46)
χ_2	414.(45)	0.80(12)	414.(26)	0.549(48)

$\lambda = 2.000, \beta = 0.335763, D = 3, L = 16$

Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	0.993(13)	0.84(12)	0.9984(61)	0.506(39)
$G(0)$	0.70715(19)	0.85(12)	0.70717(13)	0.463(29)
χ_2	450.(53)	0.84(13)	448.(23)	0.498(31)

$\lambda = 5.000, \beta = 0.265385, D = 3, L = 16$

Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	0.995(19)	1.10(18)	0.9904(62)	0.503(38)
$G(0)$	0.860804(95)	0.95(15)	0.860766(48)	0.493(22)
χ_2	546.(89)	1.04(17)	544.(28)	0.504(31)

$\lambda = 10.000, \beta = 0.241578, D = 3, L = 16$

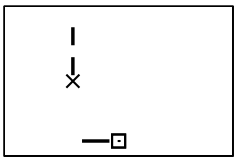
Observable	Metropolis		Worm	
	Value	τ_{int}	Value	τ_{int}
mL	1.015(26)	1.479(0.2)	0.9878(64)	0.533(41)
$G(0)$	0.935075(57)	1.403(0.2)	0.935088(22)	0.546(48)
χ_2	629.(13.)	1.76(36)	597.(31)	0.506(31)

B.2 Dynamical Behavior

In this appendix we list the data obtained in the course of the analysis of the algorithm's dynamical behavior.

B.2.1 Gaussian Case

For the Guassian case we have always compared the results of the simulations with the exact values. The deviation in units of the error is given in the last column of each table.



$D = 2, mL = 4$

$L = 8$, $m = 0.500000$

Observable	τ_{int}	Value	Δ/σ
χ_2	7.8(0.1)	8.55(0.03)	1.70
$G(0)$	3.56(0.05)	0.8202(6)	1.51
mL	8.6(0.1)	3.987(0.007)	-1.58
$\mathcal{N}/N_l$	61.(2.)	0.2857(9)	-
E	7.6(0.1)	0.3404(7)	1.70

$L = 12$, $m = 0.333333$

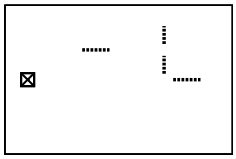
Observable	τ_{int}	Value	Δ/σ
χ_2	14.0(0.4)	18.44(0.09)	-0.55
$G(0)$	5.13(0.09)	0.932(1)	-0.66
mL	16.2(0.3)	4.00(0.01)	0.46
$\mathcal{N}/N_l$	126.(6.)	0.340(1)	-
E	13.1(0.2)	0.444(1)	-1.00

$L = 16$, $m = 0.250000$

Observable	τ_{int}	Value	Δ/σ
χ_2	19.9(0.7)	32.3(0.2)	-0.58
$G(0)$	6.6(0.1)	1.018(0.001)	-0.62
mL	26.0(0.7)	4.00(0.01)	0.35
$\mathcal{N}/N_l$	201.(13.)	0.385(1)	-
E	19.2(0.4)	0.525(1)	-0.86

$L = 24$, $m = 0.166667$

Observable	τ_{int}	Value	Δ/σ
χ_2	35.(1.)	71.8(0.6)	-1.03
$G(0)$	9.6(0.2)	1.142(0.001)	-0.46
mL	48.(1.)	4.02(0.01)	1.22
$\mathcal{N}/N_l$	427.(41.)	0.443(2)	-
E	33.(1.)	0.646(1)	-0.56



$L = 32$, $m = 0.125000$

Observable	τ_{int}	Value	Δ/σ
χ_2	66.(4.)	128.(1.)	0.01
$G(0)$	13.8(0.4)	1.231(0.002)	-0.52
mL	87.(4.)	3.98(0.02)	-0.52
$\mathcal{N}/N_l$	785.(103.)	0.491(3)	-
E	54.(2.)	0.735(2)	-0.26

$L = 48$, $m = 0.083333$

Observable	τ_{int}	Value	Δ/σ
χ_2	96.(7.)	282.(4.)	-1.40
$G(0)$	21.4(0.9)	1.358(0.003)	-0.54
mL	147.(9.)	4.06(0.03)	1.92
$\mathcal{N}/N_l$	2152.(442.)	0.546(5)	-
E	97.(5.)	0.859(3)	-0.81

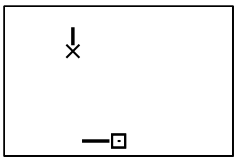
$L = 64$, $m = 0.062500$

Observable	τ_{int}	Value	Δ/σ
χ_2	191.(21.)	503.(11.)	-0.75
$G(0)$	35.(1.)	1.447(0.004)	-0.90
mL	282.(25.)	4.02(0.04)	0.51
$\mathcal{N}/N_l$	2106.(440.)	0.600(5)	-
E	187.(14.)	0.948(4)	-0.87

$D = 2, mL = 1$

$L = 8$, $m = 0.125000$

Observable	τ_{int}	Value	Δ/σ
χ_2	1369.(70.)	130.(2.)	0.80
$G(0)$	1352.(70.)	2.79(3)	0.81
mL	1355.(70.)	0.992(9)	-0.81
$\mathcal{N}/N_l$	6796.(699.)	3.18(6)	-
E	1352.(70.)	2.29(3)	0.80



$L = 12$, $m = 0.083333$

Observable	τ_{int}	Value	Δ/σ
χ_2	2074.(128.)	288.(7.)	-0.01
$G(0)$	2023.(123.)	2.88(5)	0.01
mL	2147.(134.)	1.00(1)	0.26
$\mathcal{N}/N_l$	12651.(1688.)	3.13(8)	-
E	2022.(123.)	2.38(5)	-0.00

$L = 16$, $m = 0.062500$

Observable	τ_{int}	Value	Δ/σ
χ_2	2942.(211.)	485.(15.)	-1.66
$G(0)$	2849.(203.)	2.87(6)	-1.68
mL	3206.(240.)	1.02(1)	1.40
$\mathcal{N}/N_l$	22391.(3778.)	3.0(1)	-
E	2857.(203.)	2.37(6)	-1.67

$L = 24$, $m = 0.041667$

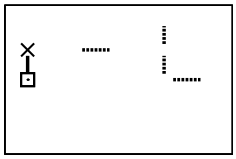
Observable	τ_{int}	Value	Δ/σ
χ_2	4138.(344.)	1204.(53.)	0.98
$G(0)$	3952.(322.)	3.19(9)	0.99
mL	4748.(419.)	0.97(2)	-1.00
$\mathcal{N}/N_l$	53426.(12661.)	3.2(1)	-
E	3957.(322.)	2.69(9)	1.00

$L = 32$, $m = 0.031250$

Observable	τ_{int}	Value	Δ/σ
χ_2	7070.(739.)	2154.(137.)	0.77
$G(0)$	6579.(668.)	3.3(1)	0.80
mL	8773.(1003.)	0.97(3)	-0.75
$\mathcal{N}/N_l$	88657.(25359.)	3.5(2)	-
E	6610.(671.)	2.8(1)	0.80

$L = 48$, $m = 0.020833$

Observable	τ_{int}	Value	Δ/σ
χ_2	12120.(1590.)	4672.(405.)	0.16
$G(0)$	11200.(1421.)	3.3(1)	0.16
mL	15158.(2181.)	0.99(4)	-0.13
$\mathcal{N}/N_l$	209499.(80090.)	3.6(4)	-
E	11220.(1423.)	2.8(1)	0.16



$L = 64$, $m = 0.015625$

Observable	τ_{int}	Value	Δ/σ
χ_2	14585.(2064.)	7785.(808.)	-0.50
$G(0)$	12438.(1648.)	3.3(1)	-0.52
mL	23189.(3963.)	1.02(5)	0.44
$\mathcal{N}/N_l$	346361.(153253.)	3.4(4)	-
E	12534.(1664.)	2.8(1)	-0.52

$D = 3, mL = 4$

$L = 8$, $m = 0.500000$

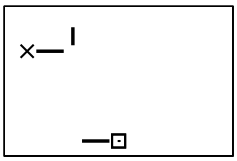
Observable	τ_{int}	Value	Δ/σ
χ_2	5.15(0.09)	12.46(0.02)	-1.02
$G(0)$	1.78(0.01)	0.6645(1)	-0.53
mL	6.31(0.08)	4.006(0.005)	1.25
$\mathcal{N}/N_l$	32.8(0.9)	0.07041(9)	-
E	6.84(0.09)	0.1713(1)	-1.02

$L = 12$, $m = 0.333333$

Observable	τ_{int}	Value	Δ/σ
χ_2	7.4(0.1)	27.44(0.07)	-0.64
$G(0)$	1.81(0.02)	0.6913(1)	0.86
mL	8.8(0.1)	4.004(0.006)	0.72
$\mathcal{N}/N_l$	49.(1.)	0.07408(7)	-
E	9.9(0.1)	0.1948(1)	0.38

$L = 16$, $m = 0.250000$

Observable	τ_{int}	Value	Δ/σ
χ_2	9.5(0.2)	48.5(0.1)	0.38
$G(0)$	1.84(0.02)	0.7061(1)	-0.09
mL	11.5(0.2)	3.995(0.006)	-0.62
$\mathcal{N}/N_l$	67.(2.)	0.07658(6)	-
E	13.1(0.2)	0.2083(1)	-0.09



$L = 24$, $m = 0.166667$

Observable	τ_{int}	Value	Δ/σ
χ_2	14.6(0.5)	108.8(0.4)	0.91
$G(0)$	1.94(0.03)	0.7224(1)	0.82
mL	17.1(0.3)	3.993(0.008)	-0.80
$\mathcal{N}/N_l$	96.(4.)	0.07934(5)	-
E	19.5(0.4)	0.22347(9)	1.16

$L = 32$, $m = 0.125000$

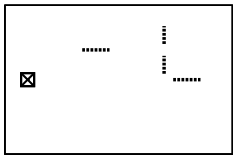
Observable	τ_{int}	Value	Δ/σ
χ_2	19.1(0.8)	192.4(0.8)	-0.01
$G(0)$	1.95(0.03)	0.73090(8)	0.71
mL	22.9(0.6)	4.001(0.009)	0.14
$\mathcal{N}/N_l$	127.(7.)	0.08083(4)	-
E	26.2(0.7)	0.23146(8)	0.29

$L = 48$, $m = 0.083333$

Observable	τ_{int}	Value	Δ/σ
χ_2	30.(2.)	433.(2.)	0.30
$G(0)$	2.07(0.04)	0.73960(7)	-0.59
mL	34.(1.)	3.99(0.01)	-0.22
$\mathcal{N}/N_l$	184.(13.)	0.08247(3)	-
E	41.(1.)	0.23989(7)	-0.49

$L = 64$, $m = 0.062500$

Observable	τ_{int}	Value	Δ/σ
χ_2	47.(4.)	769.(5.)	0.26
$G(0)$	1.94(0.04)	0.74422(6)	0.88
mL	47.(1.)	3.99(0.01)	-0.14
$\mathcal{N}/N_l$	361.(36.)	0.08336(4)	-
E	53.(2.)	0.24436(6)	0.66



$D = 3, mL = 1$

$L = 8$, $m = 0.125000$

Observable	τ_{int}	Value	Δ/σ
χ_2	104.(8.)	197.(6.)	0.81
$G(0)$	84.(6.)	1.05(0.01)	0.80
mL	34.(1.)	0.98(1)	-0.98
$\mathcal{N}/N_l$	2106.(440.)	3.20(0.02)	-
E	85.(4.)	0.55(1)	0.89

$L = 12$, $m = 0.083333$

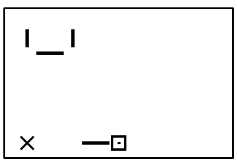
Observable	τ_{int}	Value	Δ/σ
χ_2	154.(15.)	460.(18.)	1.50
$G(0)$	110.(9.)	0.96(1)	1.59
mL	66.(7.)	0.97(1)	-1.48
$\mathcal{N}/N_l$	2106.(440.)	0.949(8)	-
E	164.(14.)	0.46(1)	1.59

$L = 16$, $m = 0.062500$

Observable	τ_{int}	Value	Δ/σ
χ_2	227.(27.)	759.(36.)	-0.23
$G(0)$	135.(12.)	0.899(8)	-0.26
mL	93.(8.)	1.00(0.02)	0.23
$\mathcal{N}/N_l$	2106.(440.)	0.400(3)	-
E	232.(23.)	0.400(8)	-0.21

$L = 24$, $m = 0.041667$

Observable	τ_{int}	Value	Δ/σ
χ_2	271.(34.)	1749.(91.)	0.23
$G(0)$	133.(12.)	0.855(6)	0.22
mL	115.(11.)	0.99(2)	-0.32
$\mathcal{N}/N_l$	2106.(440.)	0.118(1)	-
E	271.(29.)	0.355(6)	0.19



$L = 32$, $m = 0.031250$

Observable	τ_{int}	Value	Δ/σ
χ_2	344.(49.)	3127.(188.)	0.29
$G(0)$	154.(15.)	0.831(5)	0.29
mL	151.(17.)	0.98(2)	-0.62
$\mathcal{N}/N_l$	2106.(440.)	0.0500(4)	-
E	352.(42.)	0.331(5)	0.31

$L = 48$, $m = 0.020833$

Observable	τ_{int}	Value	Δ/σ
χ_2	577.(102.)	6635.(505.)	-0.55
$G(0)$	171.(18.)	0.803(3)	-0.64
mL	284.(42.)	1.01(0.03)	0.52
$\mathcal{N}/N_l$	2106.(440.)	0.0148(1)	-
E	548.(79.)	0.303(4)	-0.64

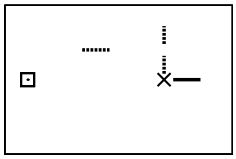
$L = 64$, $m = 0.015625$

Observable	τ_{int}	Value	Δ/σ
χ_2	751.(148.)	12459.(1026.)	0.17
$G(0)$	169.(17.)	0.794(3)	0.28
mL	304.(46.)	1.00(0.03)	0.01
$\mathcal{N}/N_l$	2106.(440.)	0.00625(5)	-
E	718.(116.)	0.294(3)	0.24

$D = 4, mL = 4$

$L = 4$, $m = 1.000000$

Observable	τ_{int}	Value	Δ/σ
χ_2	2.64(0.03)	4.496(0.006)	-0.55
$G(0)$	1.42(0.01)	0.57801(7)	0.27
mL	3.14(0.02)	4.000(0.003)	0.23
$\mathcal{N}/N_l$	14.2(0.2)	0.02733(3)	-
E	4.60(0.05)	0.08777(8)	0.40



$L = 8$, $m = 0.500000$

Observable	τ_{int}	Value	Δ/σ
χ_2	2.89(0.04)	16.47(0.02)	-0.95
$G(0)$	0.997(8)	0.60075(3)	0.04
mL	3.24(0.03)	4.003(0.002)	1.38
$\mathcal{N}/N_l$	17.0(0.3)	0.027129(9)	-
E	7.2(0.1)	0.10388(3)	-0.60

$L = 12$, $m = 0.333333$

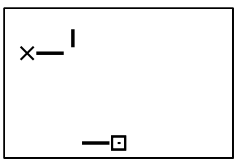
Observable	τ_{int}	Value	Δ/σ
χ_2	3.28(0.04)	36.48(0.04)	-0.28
$G(0)$	0.839(6)	0.60906(2)	0.38
mL	3.61(0.03)	3.999(0.002)	-0.26
$\mathcal{N}/N_l$	19.2(0.4)	0.028129(5)	-
E	9.2(0.1)	0.11056(1)	-0.35

$L = 16$, $m = 0.250000$

Observable	τ_{int}	Value	Δ/σ
χ_2	3.57(0.07)	64.50(0.08)	0.06
$G(0)$	0.773(7)	0.61281(1)	-0.53
mL	3.92(0.04)	3.999(0.002)	-0.16
$\mathcal{N}/N_l$	20.4(0.5)	0.028694(3)	-
E	10.5(0.1)	0.11370(1)	0.34

$L = 24$, $m = 0.166667$

Observable	τ_{int}	Value	Δ/σ
χ_2	4.10(0.09)	144.7(0.1)	1.52
$G(0)$	0.669(5)	0.616105(9)	1.23
mL	4.36(0.05)	3.995(0.002)	-1.68
$\mathcal{N}/N_l$	23.2(0.6)	0.029243(1)	-
E	12.9(0.2)	0.116505(5)	1.51



$D = 4, mL = 1$

$L = 4$, $m = 0.250000$

Observable	τ_{int}	Value	Δ/σ
χ_2	42.(1.)	65.(1.)	0.67
$G(0)$	34.(1.)	0.841(4)	0.74
mL	23.(1.)	0.993(9)	-0.71
$\mathcal{N}/N_l$	820.(188.)	0.215(3)	-
E	49.(3.)	0.344(4)	0.74

$L = 8$, $m = 0.125000$

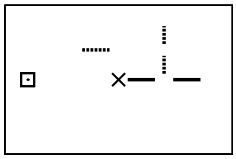
Observable	τ_{int}	Value	Δ/σ
χ_2	49.(2.)	254.(4.)	-0.43
$G(0)$	22.5(0.8)	0.672(1)	-0.43
mL	32.(2.)	1.005(0.009)	0.53
$\mathcal{N}/N_l$	2011.(657.)	0.073(1)	-
E	56.(4.)	0.172(1)	-0.29

$L = 12$, $m = 0.083333$

Observable	τ_{int}	Value	Δ/σ
χ_2	60.(3.)	577.(11.)	0.10
$G(0)$	14.3(0.4)	0.6430(4)	0.07
mL	41.(2.)	0.998(9)	-0.10
$\mathcal{N}/N_l$	1440.(414.)	0.0498(4)	-
E	62.(4.)	0.1432(5)	0.12

$L = 16$, $m = 0.062500$

Observable	τ_{int}	Value	Δ/σ
χ_2	67.(3.)	1032.(20.)	0.39
$G(0)$	8.2(0.1)	0.6328(2)	0.48
mL	49.(3.)	0.99(1)	-0.35
$\mathcal{N}/N_l$	3201.(1241.)	0.0409(3)	-
E	70.(5.)	0.1330(3)	0.48



$L = 24$, $m = 0.041667$

Observable	τ_{int}	Value	Δ/σ
χ_2	78.(4.)	2324.(46.)	0.42
$G(0)$	3.32(0.04)	0.62557(9)	0.91
mL	59.(4.)	0.99(1)	-0.44
$\mathcal{N}/N_l$	1512.(443.)	0.0348(1)	-
E	81.(7.)	0.1255(1)	0.48

B.2.2 Interacting Case, $\lambda = 0.5$

$D = 2, mL = 4$

$L = 8$, $\beta = 0.57695$

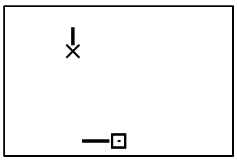
Observable	τ_{int}	Value
χ_2	2.85(4)	6.55(1)
$G(0)$	1.58(2)	0.6371(1)
mL	3.27(3)	4.003(4)
$\mathcal{N}/N_l$	16.4(2)	0.2096(2)
E	4.65(3)	0.2616(3)

$L = 12$, $\beta = 0.61567$

Observable	τ_{int}	Value
χ_2	3.13(5)	13.38(2)
$G(0)$	1.48(1)	0.6836(1)
mL	3.74(4)	3.995(5)
$\mathcal{N}/N_l$	21.5(3)	0.2429(2)
E	5.77(5)	0.3222(3)

$L = 16$, $\beta = 0.63435$

Observable	τ_{int}	Value
χ_2	3.32(5)	22.21(4)
$G(0)$	1.39(1)	0.7140(1)
mL	4.16(4)	4.008(5)
$\mathcal{N}/N_l$	27.1(4)	0.2652(2)
E	6.81(6)	0.3621(2)



$L = 22$, $\beta = 0.64927$

Observable	τ_{int}	Value
χ_2	3.49(8)	39.23(9)
$G(0)$	1.30(1)	0.7444(1)
mL	4.56(5)	4.007(6)
$\mathcal{N}/N_l$	32.4(6)	0.2891(1)
E	8.01(8)	0.4035(2)

$L = 32$, $\beta = 0.66139$

Observable	τ_{int}	Value
χ_2	3.69(9)	77.0(2)
$G(0)$	1.09(1)	0.7760(1)
mL	4.92(6)	3.991(6)
$\mathcal{N}/N_l$	41.2(9)	0.3146(1)
E	9.3(1)	0.4464(2)

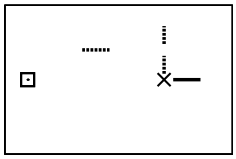
$L = 64$, $\beta = 0.67433$

Observable	τ_{int}	Value
χ_2	3.8(1)	262.1(7)
$G(0)$	0.81(1)	0.8184(1)
mL	5.9(1)	4.009(7)
$\mathcal{N}/N_l$	59.(2.)	0.3511(1)
E	12.7(3)	0.5059(1)

$D = 2, mL = 1$

$L = 8$, $\beta = 0.70071$

Observable	τ_{int}	Value
χ_2	1.54(1)	40.96(8)
$G(0)$	1.71(2)	0.9696(7)
mL	1.01(1)	0.999(4)
$\mathcal{N}/N_l$	27.7(7)	0.5608(5)
E	5.63(7)	0.723(1)



$L = 12$, $\beta = 0.69461$

Observable	τ_{int}	Value
χ_2	1.48(1)	79.8(1)
$G(0)$	1.37(1)	0.9385(5)
mL	1.10(1)	1.007(5)
$\mathcal{N}/N_l$	35.(1.)	0.5080(4)
E	5.97(8)	0.6790(8)

$L = 16$, $\beta = 0.69190$

Observable	τ_{int}	Value
χ_2	1.41(2)	129.5(3)
$G(0)$	1.17(1)	0.9231(5)
mL	1.15(1)	1.003(5)
$\mathcal{N}/N_l$	40.(1.)	0.4825(3)
E	6.01(8)	0.6583(6)

$L = 22$, $\beta = 0.69024$

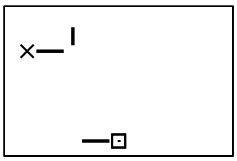
Observable	τ_{int}	Value
χ_2	1.36(2)	223.2(5)
$G(0)$	0.95(1)	0.9133(4)
mL	1.23(1)	1.003(5)
$\mathcal{N}/N_l$	45.(1.)	0.4630(2)
E	6.17(9)	0.6428(5)

$L = 32$, $\beta = 0.68899$

Observable	τ_{int}	Value
χ_2	1.35(2)	425.(1.)
$G(0)$	0.82(1)	0.9033(3)
mL	1.33(1)	1.001(6)
$\mathcal{N}/N_l$	52.(2.)	0.4472(2)
E	6.12(9)	0.6296(4)

$L = 64$, $\beta = 0.68783$

Observable	τ_{int}	Value
χ_2	1.30(2)	1404.(4.)
$G(0)$	0.639(8)	0.8943(2)
mL	1.46(2)	0.999(6)
$\mathcal{N}/N_l$	66.(2.)	0.4300(1)
E	6.08(8)	0.6157(2)



$D = 3, mL = 4$

$L = 8$, $\beta = 0.37024$

Observable	τ_{int}	Value
χ_2	2.61(4)	10.60(1)
$G(0)$	1.18(1)	0.56572(5)
mL	3.12(3)	3.998(3)
$\mathcal{N}/N_l$	17.0(2)	0.06372(4)
E	5.34(4)	0.14553(9)

$L = 12$, $\beta = 0.38303$

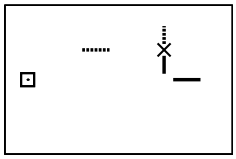
Observable	τ_{int}	Value
χ_2	2.85(4)	22.92(3)
$G(0)$	1.03(1)	0.57867(4)
mL	3.43(3)	3.997(3)
$\mathcal{N}/N_l$	20.8(3)	0.06794(3)
E	6.63(6)	0.16237(6)

$L = 16$, $\beta = 0.38831$

Observable	τ_{int}	Value
χ_2	3.13(5)	39.95(7)
$G(0)$	0.931(8)	0.58571(3)
mL	3.75(4)	3.996(3)
$\mathcal{N}/N_l$	23.9(4)	0.07062(2)
E	7.93(8)	0.17178(5)

$L = 22$, $\beta = 0.39192$

Observable	τ_{int}	Value
χ_2	3.43(8)	74.0(1)
$G(0)$	0.838(9)	0.59154(3)
mL	4.17(5)	3.999(4)
$\mathcal{N}/N_l$	27.4(5)	0.07304(1)
E	9.2(1)	0.17990(3)



$L = 32$, $\beta = 0.39439$

Observable	τ_{int}	Value
χ_2	4.0(1)	154.3(3)
$G(0)$	0.762(8)	0.59649(2)
mL	4.58(5)	3.996(4)
$\mathcal{N}/N_l$	34.7(7)	0.07523(1)
E	11.3(1)	0.18694(2)

$L = 64$, $\beta = 0.39640$

Observable	τ_{int}	Value
χ_2	4.6(1)	601.(1.)
$G(0)$	0.617(6)	0.60172(1)
mL	5.5(1)	3.996(4)
$\mathcal{N}/N_l$	45.(2.)	0.077706(5)
E	16.0(4)	0.19457(1)

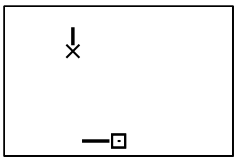
$D = 3, mL = 1$

$L = 8$, $\beta = 0.40563$

Observable	τ_{int}	Value
χ_2	2.30(3)	106.6(3)
$G(0)$	1.92(2)	0.6786(3)
mL	1.27(1)	0.995(3)
$\mathcal{N}/N_l$	52.(2.)	0.1440(1)
E	6.64(9)	0.3136(5)

$L = 12$, $\beta = 0.40175$

Observable	τ_{int}	Value
χ_2	2.30(3)	236.9(7)
$G(0)$	1.48(2)	0.6488(2)
mL	1.31(1)	0.995(3)
$\mathcal{N}/N_l$	54.(2.)	0.11721(9)
E	7.0(1)	0.2683(3)



$L = 16$, $\beta = 0.40015$

Observable	τ_{int}	Value
χ_2	2.36(4)	414.(1.)
$G(0)$	1.24(1)	0.6350(1)
mL	1.36(1)	1.002(3)
$\mathcal{N}/N_l$	60.(2.)	0.10502(6)
E	7.4(1)	0.2468(2)

$L = 22$, $\beta = 0.39907$

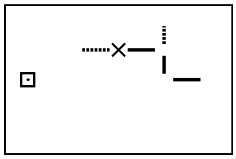
Observable	τ_{int}	Value
χ_2	2.43(5)	774.(2.)
$G(0)$	0.99(1)	0.6251(1)
mL	1.43(2)	0.999(3)
$\mathcal{N}/N_l$	70.(3.)	0.09624(4)
E	8.2(1)	0.2313(1)

$L = 32$, $\beta = 0.39833$

Observable	τ_{int}	Value
χ_2	2.72(8)	1604.(6.)
$G(0)$	0.82(1)	0.61754(9)
mL	1.56(2)	1.003(3)
$\mathcal{N}/N_l$	76.(3.)	0.08977(3)
E	9.3(1)	0.2194(1)

$L = 64$, $\beta = 0.39771$

Observable	τ_{int}	Value
χ_2	3.2(1)	6256.(24.)
$G(0)$	0.628(8)	0.61039(5)
mL	1.72(2)	0.995(3)
$\mathcal{N}/N_l$	103.(5.)	0.08367(1)
E	11.5(2)	0.20820(4)



$D = 4, mL = 4$

$L = 4$, $\beta = 0.24549$

Observable	τ_{int}	Value
χ_2	1.92(2)	4.047(4)
$G(0)$	1.31(1)	0.52102(3)
mL	2.29(2)	4.004(2)
$\mathcal{N}/N_l$	11.4(1)	0.02575(2)
E	4.54(3)	0.07906(7)

$L = 8$, $\beta = 0.27167$

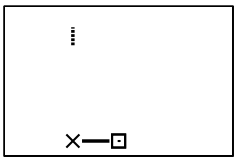
Observable	τ_{int}	Value
χ_2	1.83(2)	14.63(1)
$G(0)$	0.897(8)	0.53315(1)
mL	2.14(1)	3.998(2)
$\mathcal{N}/N_l$	13.7(1)	0.026536(7)
E	6.56(6)	0.09218(2)

$L = 12$, $\beta = 0.27763$

Observable	τ_{int}	Value
χ_2	1.93(2)	32.15(3)
$G(0)$	0.751(6)	0.53749(1)
mL	2.17(1)	3.999(2)
$\mathcal{N}/N_l$	15.5(2)	0.027699(3)
E	8.22(8)	0.09749(1)

$L = 16$, $\beta = 0.27987$

Observable	τ_{int}	Value
χ_2	2.08(3)	56.61(5)
$G(0)$	0.667(6)	0.539462(7)
mL	2.22(1)	4.002(2)
$\mathcal{N}/N_l$	17.5(2)	0.028327(2)
E	9.3(1)	0.099983(7)



$L = 24$, $\beta = 0.28156$

Observable	τ_{int}	Value
χ_2	2.17(4)	126.6(1)
$G(0)$	0.590(5)	0.541153(4)
mL	2.34(2)	4.000(1)
$\mathcal{N}/N_l$	18.2(2)	0.028921(1)
E	10.6(1)	0.102194(3)

$D = 4, mL = 1$

$L = 4$, $\beta = 0.29264$

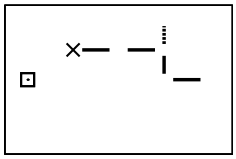
Observable	τ_{int}	Value
χ_2	3.22(5)	45.0(1)
$G(0)$	3.02(5)	0.6274(4)
mL	1.80(2)	1.003(3)
$\mathcal{N}/N_l$	51.(1.)	0.0897(1)
E	6.67(9)	0.2392(6)

$L = 8$, $\beta = 0.28506$

Observable	τ_{int}	Value
χ_2	3.70(8)	197.2(8)
$G(0)$	1.89(2)	0.5640(1)
mL	2.18(3)	1.000(2)
$\mathcal{N}/N_l$	58.(2.)	0.04490(4)
E	7.4(1)	0.1385(1)

$L = 12$, $\beta = 0.28387$

Observable	τ_{int}	Value
χ_2	4.4(1)	453.(2.)
$G(0)$	1.30(1)	0.55207(5)
mL	2.71(5)	0.998(2)
$\mathcal{N}/N_l$	61.(2.)	0.03642(1)
E	8.3(1)	0.11949(8)

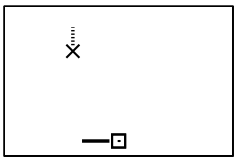


$L = 16$, $\beta = 0.28347$

Observable	τ_{int}	Value
χ_2	5.1(1)	809.(3.)
$G(0)$	1.03(1)	0.54798(3)
mL	3.24(6)	1.002(2)
$\mathcal{N}/N_l$	63.(2.)	0.03341(1)
E	9.4(1)	0.11281(4)

$L = 24$, $\beta = 0.28321$

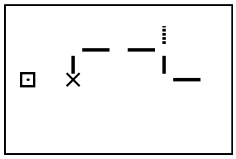
Observable	τ_{int}	Value
χ_2	5.4(2)	1862.(8.)
$G(0)$	0.74(1)	0.54518(2)
mL	3.62(7)	0.997(2)
$\mathcal{N}/N_l$	61.(2.)	0.031320(4)
E	10.2(1)	0.10830(2)

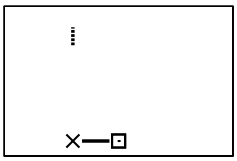


C Erratum

This is a corrected version of the thesis. It differs from the version that was reviewed by the following corrections:

- Several spelling mistakes and oversights have been corrected.
- Misuse of the term exponential growth has been corrected on pages 1, 24, 48, 55 and 69.
- Unnecessary superscript changed to subscript in (2.35).
- Clarified units used in (2.45).
- Clarified meaning of the notation $G(\mu)$ in Section 2.5.
- Corrected Equation 2.44.
- Removed superficial index in 3.6.
- Removed superficial replica-indeces in Section 3.4.
- $\tilde{c}$ is now defined explicitly in (4.16).
- Table 5.1 was corrected.





Bibliography

[1] Heinz J Rothe. *Lattice Gauge Theories: An Introduction; 3rd ed.* World Scientific Lecture Notes in Physics. World Scientific, Singapore, 2005.

[2] Ulli Wolff. Critical slowing down. *Nucl. Phys. B - Proceedings Supplements*, 17:93 – 102, 1990. ISSN 0920-5632. doi: DOI:10.1016/0920-5632(90)90224-I.

[3] Stephen L. Adler. Over-relaxation method for the monte carlo evaluation of the partition function for multiquadratic actions. *Phys. Rev. D*, 23(12):2901–2904, 1981. doi: 10.1103/PhysRevD.23.2901.

[4] Jonathan Goodman and Alan D. Sokal. Multigrid monte carlo method. conceptual foundations. *Phys. Rev. D*, 40(6):2035–2071, 1989. doi: 10.1103/PhysRevD.40.2035.

[5] Robert H. Swendsen and Jian-Sheng Wang. Nonuniversal critical dynamics in monte carlo simulations. *Phys. Rev. Lett.*, 58(2):86–88, 1987. doi: 10.1103/PhysRevLett.58.86.

[6] C. M. Fortuin and P. W. Kasteleyn. On the Random cluster model. 1. Introduction and relation to other models. *Physica*, 57:536–564, 1972. doi: 10.1016/0031-8914(72)90045-6.

[7] N. V. Prokof'ev, B. V. Svistunov, and I. S. Tupitsyn. Worm algorithm in quantum monte carlo simulations. *Phys. Lett. A*, 238(4-5):253 – 257, 1998. ISSN 0375-9601. doi: DOI:10.1016/S0375-9601(97)00957-2.

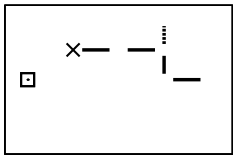
[8] Nikolay Prokof'ev and Boris Svistunov. Worm Algorithms for Classical Statistical Models. *Phys. Rev. Lett.*, 87:160601, 2001. doi: 10.1103/PhysRevLett.87.160601.

[9] Youjin Deng, Timothy M. Garoni, and Alan D. Sokal. Dynamic critical behavior of the worm algorithm for the ising model. *Phys. Rev. Lett.*, 99:110601, 2007. URL `doi:10.1103/PhysRevLett.99.071302`.

[10] Ulli Wolff. Precision check on triviality of ϕ^4 theory by a new simulation method. *Phys. Rev. D*, 2009.

[11] I. Montvay and G. Münster. *Quantum fields on a lattice.* Cambridge Monographs on Mathematical Physics. Cambridge University Press, 1994.

[12] Peter Arnold and Guy Moore. Bec transition temperature of a dilute homogeneous imperfect bose gas. *Phys. Rev. Lett.*, 87(12):120401, 2001. doi: 10.1103/PhysRevLett.87.120401.



[13] Ulli Wolff. Simulating the All-Order Strong Coupling Expansion I: Ising Model Demo. *Nucl. Phys.*, B810:491–502, 2009. doi: 10.1016/j.nuclphysb.2008.09.033.

[14] M. Lüscher and P. Weisz. Scaling Laws and Triviality Bounds in the Lattice phi**4 Theory. 1. One Component Model in the Symmetric Phase. *Nucl. Phys.*, B290:25, 1987. doi: 10.1016/0550-3213(87)90177-5.

[15] Urs M. Heller, Markus Klomfass, Herbert Neuberger, and Pavols Vranas. Numerical analysis of the higgs mass triviality bound. *Nucl, Phys. B*, 405(2-3):555 – 573, 1993. ISSN 0550-3213. doi: DOI:10.1016/0550-3213(93)90559-8.

[16] Michael Aizenman. Proof of the Triviality of phi d4 Field Theory and Some Mean-Field Features of Ising Models for d>4. *Phys. Rev. Lett.*, 47:886–886, 1981. doi: 10.1103/PhysRevLett.47.886.

[17] Michael Aizenman. Geometric Analysis of phi**4 Fields and Ising Models (Parts 1 & 2). *Commun. Math. Phys.*, 86:1, 1982. doi: 10.1007/BF01205659.

[18] Ulli Wolff. Computational Physics II, 2008. URL `http://ficus1.physik.hu-berlin.de/comphys/cp2_ws09.pdf`.

[19] Christof Gattringer and Christian B. Lang. Quantum chromodynamics on the lattice. *Lect. Notes Phys.*, 788:1–211, 2010. doi: 10.1007/978-3-642-01850-3.

[20] Ulli Wolff. Comparison between Custer Monte Carlo Algorithms in the Ising Model. *Phys. Lett.*, B228:379, 1989. doi: 10.1016/0370-2693(89)91563-3.

[21] Youjin Deng, Timothy M. Garoni, Wenan Guo, Henk W. J. Blote, and Alan D. Sokal. Cluster simulations of loop models on two-dimensional lattices. *Phys. Rev. Lett.*, 98:120601, 2007. doi: 10.1103/PhysRevLett.98.120601.

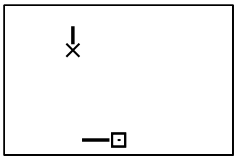
[22] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, New York, NY, USA, 2007. ISBN 0521880688, 9780521880688.

[23] Alan D. Sokal. Monte carlo methods in statistical mechanics: Foundations and new algorithms, 1996.

[24] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *J. Chem. Phys.*, 21: 1087–1092, 1953. doi: 10.1063/1.1699114.

[25] Ulli Wolff. Numerical Simulation in Quantum Field Theory, 1995. URL `http://ficus1.physik.hu-berlin.de/hera_feku/hera_feku.html`.

[26] Ulli Wolff. Monte Carlo errors with less errors. *Comput. Phys. Commun.*, 156: 143–153, 2004. doi: 10.1016/S0010-4655(03)00467-3.



[27] Ulli Wolff. Collective monte carlo updating for spin systems. *Phys. Rev. Lett.*, 62 (4):361–364, 1989. doi: 10.1103/PhysRevLett.62.361.

[28] M Galassi, Jim Davies, and James Thelier. *GNU Scientific Library: Reference Manual*. Network Theory, Bristol, 2001.

[29] Inc. Wolfram Research. *Mathematica Edition: Version 7.0*. Version 7.0. Wolfram Research, Inc., 2008.

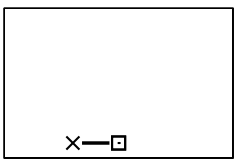
[30] Martin Lüscher. A Portable high quality random number generator for lattice field theory simulations. *Comput. Phys. Commun.*, 79:100–110, 1994. doi: 10.1016/0010-4655(94)90232-1.

[31] Andrea Pelissetto and Ettore Vicari. Critical phenomena and renormalization-group theory. *Physics Reports*, 368(6):549 – 727, 2002. ISSN 0370-1573. doi: DOI: 10.1016/S0370-1573(02)00219-3.

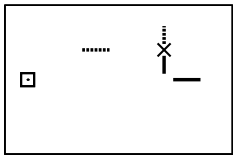
[32] U. Wolff. private communication, 2010.

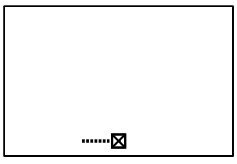
[33] Ulli Wolff. Simulating the all-order strong coupling expansion iii: O(n) sigma/loop models. *Nucl. Phys. B*, 824(1-2):254 – 272, 2010. ISSN 0550-3213. doi: DOI: 10.1016/j.nuclphysb.2009.09.006.

[34] Xuepeng Sun. Monte carlo studies of three-dimensional o(1) and o(4) $\varphi 4$ theory related to bose-einstein condensation phase transition temperatures. *Phys. Rev. E*, 67(6):066702, 2003. doi: 10.1103/PhysRevE.67.066702.

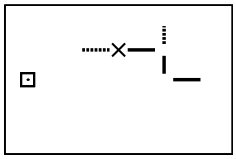


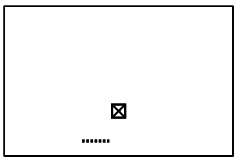
List of Figures





List of Tables





Acknowledgements

First of all, I would like to thank Prof. Wolff for introducing me to this interesting field, for sharing his knowledge and experience and for patiently discussing my questions with me. I owe thanks to all members of the group Computational Physics, being a part of which I enjoyed very much. Particularly I am indebted to Oliver Baer for his lectures and discussions, to Tomasz Korzec for always finding some time, to Fatih Tekin for talking about scales in a cold night and to Andreas Furchner and Florian Wende for keeping me company along the way and for fruitful discussions. I also enjoyed sharing views with my fellow Hessian Stefan Schaefer.

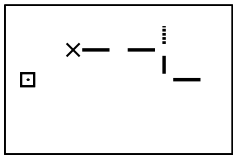
I owe special thanks to Willi Rath, Marina Marinkovic, Andreas Furchner and Fatih Tekin for spending their free time conducting aerodynamical experiments on the Zeuthener See with me. I hope this collaboration will continue in the future.

My proof-readers Andreas Furchner, Florian Wende, Tomasz Korzec, Henrik Grundmann and Eva Baer have provided invaluable services. Thank you for your support.

Henrik Grundmann and Uwe Specht, I have enjoyed learning about physics with you. Thank you for sharing with me the first four years of our studies.

I would also like to thank my parents for literally everything, and my brother Helge for reminding me of the general awsomeness of physics whenever I lost sight of it while searching a wrong sign. I am grateful to my friends Markus und Marie for keeping me virtual company during long nights in Adlershof.

Finally, I want to thank my love Grietje for enduring my alternating absence and absent-mindedness and for her unwavering support. As long as you are with me, I can not wait to see tomorrow.



Selbständigkeitserklärung

Ich erkläre, dass ich die vorliegende Arbeit selbständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe.

Berlin, den 12.05.2010 Ingmar Vierhaus